Aalborg Universitet

AALBORG UNIVERSITY
DENMARK

Single-Microphone Speech Enhancement and Separation Using Deep Learning

Kolbæk, Morten

DOI (link to publication from Publisher):
10.54337/aau300036831

Publication date:
2018

Document Version
Publisher's PDF, also known as Version of record

Link to publication from Aalborg University

Citation for published version (APA):
Kolbæk, M. (2018). *Single-Microphone Speech Enhancement and Separation Using Deep Learning*. Aalborg Universitetsforlag. Ph.d.-serien for Det Tekniske Fakultet for IT og Design, Aalborg Universitet https://doi.org/10.54337/aau300036831

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal -

Take down policy
If you believe that this document breaches copyright please contact us at vbn@aub.aau.dk providing details, and we will remove access to the work immediately and investigate your claim.

Downloaded from vbn.aau.dk on: May 05, 2023

SINGLE-MICROPHONE SPEECH ENHANCEMENT AND SEPARATION USING DEEP LEARNING

BY
MORTEN KOLBÆK

THESIS SUBMITTED 2018

AALBORG UNIVERSITY
DENMARK

Single-Microphone Speech Enhancement and Separation Using Deep Learning

PhD Thesis

Morten Kolbæk

2018

Thesis submitted:	August 31, 2018
PhD supervisor:	Professor Jesper Jensen Aalborg University, Denmark
Assistant PhD supervisor:	Professor Zheng-Hua Tan Aalborg University, Denmark
PhD committee:	Associate Professor Thomas Arildsen (chairman) Aalborg University, Denmark
	Professor Reinhold Häb-Umbach Paderborn University, Germany
	Professor John H.L. Hansen The University of Texas at Dallas, USA
PhD Series:	Technical Faculty of IT and Design, Aalborg University
Department:	Department of Electronic Systems

ISSN (online): 2446-1628
ISBN (online): 978-87-7210-256-6

Published by:
Aalborg University Press
Langagervej 2
DK – 9220 Aalborg Ø
Phone: +45 99407140
aauf@forlag.aau.dk
forlag.aau.dk

© Copyright: Morten Kolbæk, except where otherwise stated.

Printed in Denmark by Rosendahls, 2018

About the Author

Morten Kolbæk

Morten Kolbæk received the B.Eng. degree in electronic design at Aarhus University, Business and Social Sciences, AU Herning, Denmark, and the M.Sc. degree in signal processing and computing from Aalborg University, Denmark, in 2013 and 2015, respectively. He is currently pursuing his PhD degree at the section for Signal and Information Processing at the Department of Electronic Systems, Aalborg University, Denmark, under the supervision of Professor Jesper Jensen and Professor Zheng-Hua Tan. His main research interests include single-microphone algorithms for speech enhancement and multi-talker speech separation, machine learning, deep learning in particular, and intelligibility improvement of noisy speech for hearing aids applications.

This page intentionally left blank.

Abstract

The cocktail party problem comprises the challenging task of listening to and understanding a speech signal in a complex acoustic environment, where multiple speakers and background noise signals simultaneously interfere with the speech signal of interest. A signal processing algorithm that can effectively increase the speech intelligibility and quality of speech signals in such complicated acoustic situations is highly desirable. Especially for applications involving mobile communication devices and hearing assistive devices, increasing speech intelligibility and quality of noisy speech signals has been a goal for scientists and engineers for more than half a century. Due to the re-emergence of machine learning techniques, today, known as deep learning, the challenges involved with such algorithms might be overcome.

In this PhD thesis, we study and develop deep learning-based techniques for two major sub-disciplines of the cocktail party problem: *single-microphone speech enhancement* and *single-microphone multi-talker speech separation*.

Specifically, we conduct in-depth empirical analysis of the generalizability capability of modern deep learning-based single-microphone speech enhancement algorithms. We show that performance of such algorithms is closely linked to the training data, and good generalizability can be achieved with carefully designed training data. Furthermore, we propose utterance-level Permutation Invariant Training (uPIT), a deep learning-based algorithm for single-microphone speech separation and we report state-of-the-art results on a speaker-independent multi-talker speech separation task. Additionally, we show that uPIT works well for joint speech separation and enhancement without explicit prior knowledge about the noise type or number of speakers, which, at the time of writing, is a capability only shown by uPIT. Finally, we show that deep learning-based speech enhancement algorithms designed to minimize the classical short-time spectral amplitude mean squared error leads to enhanced speech signals which are essentially optimal in terms of Short-Time Objective Intelligibility (STOI), a state-of-the-art speech intelligibility estimator. This is important as it suggests that no additional improvements in STOI can be achieved by a deep learning-based speech enhancement algorithm, which is designed to maximize STOI.

This page intentionally left blank.

Resumé

Cocktailparty-problemet beskriver udfordringen ved at forstå et talesignal i et komplekst akustisk miljø, hvor stemmer fra adskillige personer, samtidig med baggrundsstøj, interferer med det ønskede talesignal. En signalbehandlings algoritme, som effektivt kan øge taleforståeligheden eller talekvaliteten af støjfyldte talesignaler, er yderst eftertragtet. Specielt indenfor applikationer som vedrører mobil kommunikation eller høreapparater, har øgning af taleforståelighed eller talekvalitet af støjfyldte talesignaler været et mål for videnskabsfolk og ingeniører i mere end et halvt århundrede. Grundet en genopstået interesse for maskinlærings teknikker, som i dag er kendt som dyb læring, kan nogle af de udfordringer som er forbundet med sådanne algoritmer, måske blive løst.

I denne Ph.d.-afhandling studerer og udvikler vi dyb-læringsbaserede teknikker anvendeligt for to store underdiscipliner af cocktailparty-problemet: *enkelt-mikrofon taleforbedring* og *enkelt-mikrofon multi-taler taleseparation.*

Specifikt foretager vi dybdegående empiriske analyser af generaliseringsegenskaberne af moderne dyb-læringsbaserede enkelt-mikrofons taleforbedringsalgoritmer. Vi viser at ydeevnen af disse algoritmer er tæt forbundet med mængden og kvaliteten af træningsdata, og gode generaliseringsegenskaber kan opnås ved omhyggeligt designet træningsdata. Derudover præsenterer vi utterance-level Permutation Invariant Training (uPIT), en dyb læringsbaseret algoritme til enkelt-mikrofon taleseparation og vi rapporterer state-of-the-art resultater for en taler-uafhængig multi-taler taleseparationsopgave. Ydermere viser vi, at uPIT fungerer godt til både taleseparation samt taleforbedring samtidigt, hvilket på tidspunktet for denne afhandling, er en egenskab, som kun uPIT har. Endelig viser vi, at dyb-læringsbaserede taleforbedrings algoritmer som er designet til at maksimere den klassiske short-time spectral amplitude mean squared error fører til forbedrede talesignaler, som essentielt er optimale med hensyn til Short-Time Objective Intelligibility (STOI), en state-of-the-art taleforståelighedsprædiktor. Dette er vigtig, da det antyder at ingen yderligere forbedring af STOI kan opnås selv med dyb-læringsbaserede taleforbedrings algoritmer, som er designet til at maksimere STOI.

This page intentionally left blank.

Contents

This page intentionally left blank.

List of Abbreviations

ADFD	Akustiske Databaser for Dansk
AMS	Amplitude Modulation Spectrogram
AM	Amplitude Mask
ANN	Artificial Neural Network
ASA	Auditory Scene Analysis
ASR	Automatic Speech Recognition
BLSTM	Bi-directional Long Short-Term Memory
BSS	Blind-Source Separation
CASA	Computational Auditory Scene Analysis
CC	Closed-Condition
CNN	Convolutional Neural Network
CNTK	Microsoft Cognitive Toolkit
DANet	Deep Attractor Network
DBN	Deep Belief Network
DFT	Discrete Fourier Transform
DL	Deep Learning
DNN	Deep Neural Network
DPCL	Deep Clustering
DRNN	Deep Recurrent Neural Network
DTFT	Discrete-Time Fourier Transform
EER	Equal Error Rate
ELC	Envelope Linear Correlation
EMSE	Envelope Mean Squared Error
ERB	Equivalent Rectangular Bandwidth

ESTOI	Extended Short-Time Objective Intelligibility
EVD	Eigen-Value Decomposition
FIR	Finite Impulse Response
FNN	Feed-forward Neural Network
GFE	Gammatone Filter bank Energies
GMM	Gaussian Mixture Model
HMM	Hidden Markov Model
IAM	Ideal Amplitude Mask
IBM	Ideal Binary Mask
IDFT	Inverse Discrete Fourier Transform
IIR	Infinite Impulse Response
INPSM	Ideal Non-negative Phase Sensitive Mask
IPSF	Ideal Phase Sensitive Filter
IPSM	Ideal Phase Sensitive Mask
IRM	Ideal Ratio Mask
KLT	Karhunen-Loève Transform
LPC	Linear Predictive Coding
LSTM	Long Short-Term Memory
MFCC	Mel-Frequency Cepstrum Coefficient
MLP	Multi-Layer Perceptron
MMELC	Maximum Mean Envelope Linear Correlation
MMSE	Minimum Mean Squared Error
MOS	Mean Opinion Score
MRF	Markov Random Field
MSE	Mean Squared Error
NMF	Non-negative Matrix Factorization
OC	Open-Condition
PDF	Probability Density Function
PESQ	Perceptual Evaluation of Speech Quality
PIT	Permutation Invariant Training
PSA	Phase Sensitive Approximation

PSD	Power Spectral Density
PSF	Phase Sensitive Filter
PSM	Phase Sensitive Mask
RASTA-PLP	Relative Spectral Transform - Perceptual Linear Prediction
RBM	Restricted Boltzmann Machine
RMS	Root Mean Square
RNN	Recurrent Neural Network
ROC	Receiver Operating Characteristics
ReLU	Rectified Linear Unit
SAR	Source-to-Artifact Ratio
SDR	Source-to-Distortion Ratio
SE	Speech Enhancement
SGD	Stochastic Gradient Descent
SIR	Source-to-Interference Ratio
SI	Speech Intelligibility
SNR	Signal-to-Noise Ratio
SQ	Speech Quality
SR	Speaker Recognition
SSN	Speech Shaped Noise
STFT	Short-Time Fourier Transform
STOI	Short-Time Objective Intelligibility
STSA	Short-Time Spectral Amplitude
SVD	Singular-Value Decomposition
SVM	Support Vector Machine
SV	Speaker Verification
T-F	Time-Frequency
UBM	Universal Background Model
VAD	Voice Activity Detection
WSJ0	Wall Street Journal
WGN	White Gaussian Noise
uPIT	utterance-level Permutation Invariant Training

This page intentionally left blank.

List of Publications

This main body (Part II) of this thesis consists of the following publications:

[A] M. Kolbæk, Z. H. Tan, and J. Jensen, "Speech Intelligibility Potential of General and Specialized Deep Neural Network Based Speech Enhancement Systems", *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 1, pp. 153–167, January 2017.

[B] M. Kolbæk, Z.-H. Tan, and J. Jensen, "Speech Enhancement Using Long Short-Term Memory Based Recurrent Neural Networks for Noise Robust Speaker Verification", *IEEE Spoken Language Technology Workshop*, pp. 305-311, December 2016.

[C] D. Yu, M. Kolbæk, Z.-H. Tan, and J. Jensen, "Permutation Invariant Training of Deep Models for Speaker-Independent Multi-Talker Speech Separation", *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pp 241-245, March 2017.

[D] M. Kolbæk, D. Yu, Z.-H. Tan, and J. Jensen, "Multi-Talker Speech Separation With Utterance-Level Permutation Invariant Training of Deep Recurrent Neural Networks", *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 10, pp. 1901-1913, October 2017.

[E] M. Kolbæk, D. Yu, Z.-H. Tan, and J. Jensen, "Joint Separation and Denoising of Noisy Multi-Talker Speech Using Recurrent Neural Networks and Permutation Invariant Training", *IEEE International Workshop on Machine Learning for Signal Processing*, pp. 1-6, September 2017.

[F] M. Kolbæk, Z.-H. Tan, and J. Jensen, "Monaural Speech Enhancement Using Deep Neural Networks by Maximizing a Short-Time Objective Intelligibility Measure", *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pp 5059-5063, April 2018.

[G] M. Kolbæk, Z.-H. Tan, and J. Jensen, "On the Relationship between Short-Time Objective Intelligibility and Short-Time Spectral-Amplitude Mean Squared Error for Speech Enhancement", *under major revision in IEEE/ACM Transactions on Audio, Speech, and Language Processing*, August 2018.

This page intentionally left blank.

Preface

This thesis documents the scientific work carried out as part of the PhD project entitled *Single-Microphone Speech Enhancement and Separation Using Deep Learning*. The thesis is submitted to the Technical Doctoral School of IT and Design at Aalborg University in partial fulfillment of the requirements for the degree of Doctor of Philosophy. The project was funded by the Oticon Foundation[1], and the work was carried out in the period from September 2015 to August 2018 within the Signal and Information Processing Section, in the Department of Electronic Systems, at Aalborg University. Parts of the work was carried out during a four-month secondment at the Interactive Systems Design Lab at the University of Washington, Seattle USA, and at Microsoft Research, Redmond USA.

The thesis is structured in two parts: a general introduction and a collection of scientific papers. The introduction review classical algorithms and deep learning-based algorithms for single-microphone speech enhancement and separation, and furthermore summarizes the scientific contributions of the PhD project. The introduction is followed by a collection of seven papers that are published in or submitted to peer-reviewed journals or conferences.

I would like to express my deepest gratitude to my two supervisors Jesper Jensen and Zheng-Hua Tan for their support and guidance throughout the project. In particular, I would like to thank Jesper Jensen for his sincere dedication to the project and for his abundant, and seemingly endless, supply of constructive criticism, which, although daunting at times, unarguably has improved all aspects of my work. Furthermore, I would like to give a special thanks to Dong Yu for a very giving and pleasant collaboration for which I am very grateful. Also, I would like to thank Les Atlas, Scott Wisdom, Tommy Powers and David Dolengewicz from the Interactive Systems Design Lab for their hospitality and helpfulness during my stay at University of Washington. Last, but not least, I wish to thank my family for their unconditional support.

Morten Kolbæk
Bjerghuse, July 17, 2018

[1] `http://www.oticonfoundation.com`

This page intentionally left blank.

Part I

Introduction

This page intentionally left blank.

Introduction

Most of us take it for granted and use it effortless on a daily basis; our ability to speak and hear. Nevertheless, the human speech production- and auditory systems are truly unique [1–7].

We are probably all familiar with the challenging situation at a dinner party when you attempt to converse with the person sitting across the table. Other persons, having their own conversations, are sitting around you, and you have to concentrate to hear the voice of the person you are trying to have a conversation with. Remarkably, the more you concentrate on the voice of your conversational partner, the more you understand and the less you feel distracted by the people talking loudly around you. This ability of selective auditory attention is one of the astonishing capabilities of the human auditory system. In fact, in 1953 it was proposed as an engineering discipline in the academic literature by Colin Cherry [8] when he asked:

> *How do we recognize what one person is saying when others are speaking at the same time (the "cocktail party problem")? On what logical basis could one design a machine ("filter") for carrying out such an operation?*
>
> *– Colin Cherry, 1953.*

Ever since Colin Cherry coined the term *cocktail party problem*, it has been, and still is, a very active topic of research within multiple scientific disciplines such as psychoacoustics, auditory neuroscience, electrical engineering, and computer science [4, 9–17], and although Colin Cherry studied speech-interference signals in his seminal work in 1953, today, the cocktail party problem encompasses both speech and non-speech-interference signals [18, 19].

In this PhD thesis, we study aspects of the cocktail party problem. Specifically, motivated by a re-emergence of a branch of machine learning, today, commonly known as *deep learning* [20], we investigate how deep learning techniques can be used to address some of the challenges in two major subdisciplines of the cocktail party problem: *single-microphone speech enhancement* and *single-microphone multi-talker speech separation*.

1 Speech Enhancement and Separation

The common goal of single-microphone speech enhancement and single-microphone multi-talker speech separation algorithms is to improve some aspects, e.g. quality or intelligibility, of a single-microphone recording of one or more degraded speech signals [11, 21–23]. As the name implies, single-microphone algorithms process sound signals captured by a single microphone. Such algorithms are useful in applications where microphone arrays cannot be utilized, e.g. due to space, power, or hardware-cost restrictions, e.g. for in-the-ear hearing aids. Furthermore, since single-microphone algorithms do not rely on the spatial locations of target and interference signals, single-microphone algorithms compliment multi-microphone algorithms and can be used as a post-processing step for techniques such as beamforming, as those techniques are mainly effective, when target and interference signals are spatially separated [24]. Therefore, algorithms capable of enhancing or separating speech signals from single-microphone recordings are highly desirable.

The main difference between speech enhancement and multi-talker speech separation algorithms is the number of target signals. If the target is only a single speech signal and all remaining sounds in the recording, both speech and non-speech sounds, are considered as noise, extracting that particular speech signal from the recording is considered as a speech enhancement task. On the other hand, if the recording contains multiple speech signals, and possibly multiple non-speech sounds, and two or more of these speech signals are of interest, the task is a multi-talker speech separation task. In this sense, the speech enhancement problem may be seen as a special case of the multi-talker speech separation problem.

Applications for speech enhancement include mobile communication devices, e.g. mobile phones, or hearing assistive devices where usually only a single speech signal is the target. For these applications, successful algorithms have been developed, which e.g. rely on interference characteristics which are different than speech. Hence, these methods would not perform well for speech-like interference signals. Applications for multi-talker speech separation include automatic meeting transcription, multi-party human-machine interaction, e.g. for video games like Xbox or PlayStation, or automatic captioning for audio/video recordings, e.g. for YouTube or Facebook, all situations where overlapping speech is not uncommon. Since the interference signals for these applications are speech signals, single-microphone multi-talker speech separation possesses additional challenges compared to single-microphone speech enhancement. However, in theory, a perfect system for multi-talker speech separation would also be a perfect system for speech enhancement, but not the other way around.

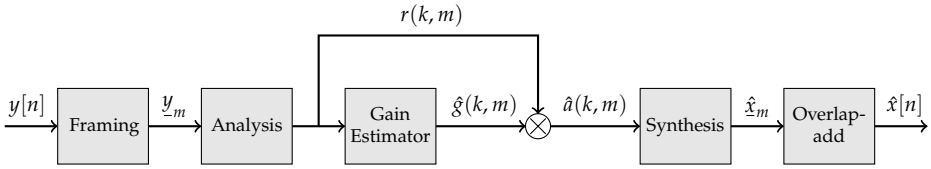


Fig. 1: Classical gain-based speech enhancement system. The noisy time-domain signal $y[n] = x[n] + v[n]$ is first segmented into overlapping frames $\underline{y}_m$. An analysis stage then applies a transform to arrive in a transform-domain $r(k,m)$ for time-frame m and transform-coefficient k. A gain $\hat{g}(k,m)$ is then estimated and applied to $r(k,m)$ to arrive at an enhanced transform-coefficient $\hat{a}(k,m) = \hat{g}(k,m)r(k,m)$. Finally, a synthesis stage transforms the enhanced transform-coefficient into time domain and the final time-domain signal $\hat{x}[n]$ is obtained by overlap-add.

1.1 Classical Speech Enhancement Algorithms

Let $x[n]$ be a sample of a clean time-domain speech signal and let a noisy observation $y[n]$ be defined as

$$y[n] = x[n] + v[n], \tag{1}$$

where $v[n]$ is an additive noise sample representing any speech and non-speech, interference signal. Then, the goal of single-microphone speech enhancement is to acquire an estimate $\hat{x}[n]$ of $x[n]$, which in some sense is "close to" $x[n]$ using $y[n]$ only.

Throughout the years, a wide range of techniques have been proposed for estimating $x[n]$ and many of these techniques follow the gain-based approach shown in Fig. 1 , e.g. [22, 23]. First, the noisy time-domain signal $y[n]$ is segmented into overlapping frames $\underline{y}_m$ using a sliding window of length N. An analysis stage then applies a transform, e.g. the Discrete Fourier Transform (DFT), to the frames to arrive in a transform-domain $r(k,m)$ for time-frame m and transform-coefficient k. An estimator, to be further defined in the next sections, estimates a gain value $\hat{g}(k,m)$ that is applied to $r(k,m)$ to arrive at an enhanced transform-coefficient $\hat{a}(k,m) = \hat{g}(k,m)r(k,m)$. A synthesis stage then applies an inverse transform to the enhanced transform-coefficients to transform the coefficients back to time domain. Finally, the time-domain signal $\hat{x}[n]$ is obtained by overlap-adding the enhanced time-domain frames $\underline{\hat{x}}_m$ [25].

Although many speech enhancement algorithms follow the gain-based approach, their strategy for finding the gain value $\hat{g}(k,m)$, i.e. the design of the gain estimator, can be very different, and, in general, these techniques may be divided into four classes [22]: 1) Spectral subtractive-based algorithms (Sec. 1.1.1), 2) Statistical model-based algorithms (Sec. 1.1.2), 3) Subspace based algorithms (Sec. 1.1.3), and 4) Machine learning-based algorithms (Sec. 1.1.4).

1.1.1 Spectral Subtraction Methods

Speech enhancement algorithms based on spectral subtraction belong to the first class of algorithms proposed for speech enhancement and were developed in the late 1970s [22, 26, 27]. Specifically, let $y(k,m)$, $x(k,m)$, and $v(k,m)$ be the Short-Time Fourier Transform (STFT) coefficients of the noisy signal $y[n]$, clean signal $x[n]$, and noise signal $v[n]$, from Eq. (1), respectively. The spectral subtraction algorithm in its simplest form is then defined as

$$\hat{x}(k,m) = [|y(k,m)| - |v(k,m)|]\, e^{j\phi_y(k,m)}, \tag{2}$$

where $|\cdot|$ denotes absolute value and $e^{j\phi_y(k,m)}$ is the phase of the noisy STFT coefficients $y(k,m)$. From Eq. (2) it is clear why this algorithm is named "spectral subtraction" as the estimate $\hat{x}(k,m)$ is acquired simply by subtracting the noise magnitude $|v(k,m)|$ from the magnitude of the noisy signal $|y(k,m)|$ and appending the noisy phase $e^{j\phi_y(k,m)}$. Furthermore, by slightly rewriting Eq. (2), we arrive at

$$\hat{x}(k,m) = g(k,m)|y(k,m)|e^{j\phi_y(k,m)}, \tag{3}$$

where

$$g(k,m) = 1 - \frac{|v(k,m)|}{|y(k,m)|} \tag{4}$$

is the gain function, which clearly shows that spectral subtraction as defined by Eq. (2) indeed belongs to the family of gain-based speech enhancement algorithms. Finally, although spectral subtraction as defined by Eq. (2) was primarily motivated heuristically [26], it was later shown [27] that Eq. (2) is closely related to the maximum likelihood estimate of the clean speech Power Spectral Density (PSD), when speech and noise are modeled as independent stochastic processes [27]. An assumption that is used heavily in later successful speech enhancement algorithms [23, 28].

Although speech enhancement algorithms based on the spectral subtraction principle effectively reduce the noise in noisy speech signals, it has a few disadvantages. First, it requires an accurate estimate of the noise magnitude $|v(k,m)|$, which in general is not easily available and might be time varying. As a consequence, $|v(k,m)|$ was first estimated from non-speech periods prior to speech activity, e.g. using a Voice Activity Detection (VAD) algorithm [22]. Furthermore, due to estimation errors of $|v(k,m)|$, $|\hat{x}(k,m)|$ might be negative, which by definition is an invalid magnitude spectrum. Several techniques have been proposed to alleviate this side-effect (e.g. [22, 26, 29–31]) and the simplest is to apply a half-wave rectifier and set all negative values to zero. Another technique is to set negative values to the value of adjacent non-negative frames, but regardless of the technique, spectral subtractive-based techniques are prone to signal distortions known as *musical noise* due to estimation errors in the estimate of the noise magnitude $|v(k,m)|$.

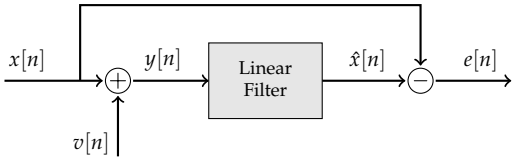


Fig. 2: Linear estimation problem for which Wiener filters are optimal in a mean squared error sense.

1.1.2 Statistically Optimal Methods

Although spectral subtractive-based techniques are effective speech enhancement algorithms, they are primarily based on heuristics and not derived deliberately to be mathematically optimal. If, however, the speech enhancement problem is formulated as a statistical estimation problem with a well-defined optimality criterion and strictly defined statistical assumptions, a class of *optimal* speech enhancement algorithms can be developed [21–23, 27, 28, 32–38]. One such class is the Minimum Mean Squared Error (MMSE) estimators, for which two large sub-classes are the linear MMSE estimators, commonly known as *Wiener filters* after the mathematician Nobert Wiener [39], and the non-linear Short-Time Spectral Amplitude (STSA)-MMSE estimators [28].

Basic Wiener Filters

Wiener filters are minimum mean squared error optimal linear filters for the linear estimation problem shown in Fig. 2, where the observed signal $y[n]$ is given by $y[n] = x[n] + v[n]$, where $x[n]$ and $v[n]$ are assumed to be uncorrelated and stationary stochastic processes [21, 22, 33]. Wiener filters can have either a Finite Impulse Response (FIR) or an Infinite Impulse Response (IIR) or be even non-causal. For the causal FIR Wiener filter, the estimated signal $\hat{x}[n]$ is given by

$$\hat{x}[n] = \underline{h}_o^T \underline{y}(n), \tag{5}$$

where

$$\underline{h}_o = [h_1,\ h_2,\ \ldots,\ h_L]^T \tag{6}$$

are the optimal filter coefficients and

$$\underline{y}(n) = [y[n],\ y[n-1],\ \ldots,\ y[n-L+1]]^T \tag{7}$$

are the past L samples of the observed signal. The optimal filter, $\underline{h}_o$, i.e. the Wiener filter, is then defined as

$$\underline{h}_o = \arg\min_{\underline{h}}\ J_x(\underline{h}), \tag{8}$$

where $J_x(\underline{h})$ is the mean squared error given by

$$J_x(\underline{h}) = \mathbb{E}\{e^2[n]\} = \mathbb{E}\{(x[n] - \hat{x}[n])^2\}, \tag{9}$$

and $\mathbb{E}\{\cdot\}$ denote mathematical expectation. Finally, by differentiating Eq. (9) with respect to $\underline{h}$, equating to zero, and solving for $\underline{h}$, the optimal filter coefficients $\underline{h}_o$ are found to be

$$\underline{h}_o = (\underline{\underline{R}}_{\underline{x}\underline{x}} + \underline{\underline{R}}_{\underline{v}\underline{v}})^{-1} \underline{r}_{x\underline{x}}, \tag{10}$$

which is the well-known Wiener-Hopf solution[2] [22, 40], where $\underline{\underline{R}}_{\underline{x}\underline{x}}$ and $\underline{\underline{R}}_{\underline{v}\underline{v}}$ denote the autocorrelation matrices of $\underline{x}$ and $\underline{v}$, respectively, and $\underline{r}_{x\underline{x}} = \mathbb{E}\{x[n]\underline{x}\}$ denote the autocorrelation vector. From Eq. (10) it is seen that the optimal filter coefficients $\underline{h}_o$ are based on $\underline{\underline{R}}_{\underline{x}\underline{x}}$, $\underline{\underline{R}}_{\underline{v}\underline{v}}$, and $\underline{r}_{x\underline{x}}$, which are not directly available and must be estimated, for the filter to be used in practice. Since the noise process $v[n]$ is assumed to be stationary, accurate estimates of $\underline{\underline{R}}_{\underline{v}\underline{v}}$ might be acquired during non-speech periods and used during speech-active periods [21, 22].

An alternative to the time-domain Wiener filter is the frequency-domain Wiener filter. If the filter $\underline{h}$ is allowed to be of infinite duration and non-causal, i.e. $\underline{h}' = [\ldots, h'_{-1}, h'_0, h'_1, \ldots]$, the Wiener filter can be defined in the frequency domain using a similar approach as just described. Let

$$\hat{x}(\omega) = g(\omega)y(\omega), \tag{11}$$

where $\hat{x}(\omega)$, $g(\omega)$, and $y(\omega)$ denote the Discrete-Time Fourier Transform (DTFT) of the estimated speech signal $\hat{x}[n]$, the infinite duration time-domain filter $\underline{h}'$, and the noisy speech signal $y[n]$, respectively. The frequency domain Wiener filter is then given as [21, 22]

$$H(\omega) = \frac{P_x(\omega)}{P_x(\omega) + P_v(\omega)}, \tag{12}$$

where $P_x(\omega)$, and $P_v(\omega)$ are the PSD of the clean speech signal $x[n]$, and noise signal $v[n]$, respectively. Alternatively, the frequency domain Wiener filter can be formulated as

$$H(\omega) = \frac{\xi_\omega}{\xi_\omega + 1}, \tag{13}$$

where

$$\xi_\omega = \frac{P_x(\omega)}{P_v(\omega)} \tag{14}$$

[2]The Wiener-Hopf solution is usually on the form $\underline{\underline{R}}_{\underline{y}\underline{y}}^{-1} \underline{r}_{x\underline{y}}$ but since $x[n]$ and $v[n]$ are assumed uncorrelated, $\underline{\underline{R}}_{\underline{y}\underline{y}} = \underline{\underline{R}}_{\underline{x}\underline{x}} + \underline{\underline{R}}_{\underline{v}\underline{v}}$ and $\underline{r}_{x\underline{y}} = \underline{r}_{x\underline{x}}$.

is known as the *a priori* Signal-to-Noise Ratio (SNR) at frequency ω. From Eqs. (12) and (13) it is seen that the frequency-domain Wiener filter $g(\omega)$ is real, even, and non-negative and, consequently, does not modify the phase of $y(\omega)$, hence $\hat{x}(\omega)$ will have the same phase as $y(\omega)$, similarly to the spectral subtractive-based approaches [41]. Furthermore, from Eq. (13) it can be deduced that the Wiener filter operates by suppressing signals with low SNR relatively more than signals with higher SNR. Finally, similarly to the time-domain Wiener filter, the frequency-domain Wiener filter, as formulated by Eqs. (12) and (13), is not directly applicable in practice as speech may only be stationary during short time periods and information about the *a priori* SNR is not available in general. Consequently, $P_x(\omega)$ and $P_v(\omega)$ must be estimated using e.g. iterative techniques for short time periods where speech and noise are approximately stationary, e.g. [21, 22].

Basic STSA-MMSE Estimators

Although the Wiener filter is considered the optimal complex spectral estimator, it is not the optimal spectral amplitude estimator, and based on the common belief at-the-time that phase was much less important than amplitude for speech enhancement (see e.g. [41–46] and references therein), it led to the development of optimal spectral amplitude estimators, commonly known as STSA-MMSE estimators [28].

Differently from the Wiener filters, STSA-MMSE estimators do not assume a linear relation between the observed data and the estimator. Instead, the STSA-MMSE estimators are derived using a Bayesian statistical framework, where explicit assumptions are made about the probability distributions of speech and noise DFT coefficients.

Specifically, let $A(k,m)$, and $R(k,m)$, $k = 1, 2, \ldots, K$, $m = 1, 2, \ldots, M$ denote random variables representing the K-point STFT magnitude spectra for time frame m of the clean speech signal $x[n]$, and noisy speech signal $y[n]$, respectively. Let $\hat{A}(k,m)$, and $V(k,m)$ be defined in a similar manner for the estimated speech signal $\hat{x}[n]$ and the noise signal $v[n]$, respectively. In the following the frame index m will be omitted for convenience as all further steps apply for all time frames. Let

$$\underline{A} = [A_1, A_2, \ldots, A_K]^T, \tag{15}$$

$$\underline{R} = [R_1, R_2, \ldots, R_K]^T, \tag{16}$$

and

$$\underline{\hat{A}} = \left[\hat{A}_1, \hat{A}_2, \ldots, \hat{A}_K\right]^T, \tag{17}$$

be the stack of these random variables into random vectors. Also, let $p(\underline{A}, \underline{R})$ denote the joint Probability Density Function (PDF) of clean and noisy spectral magnitudes and $p(\underline{A}|\underline{R})$, and $p(\underline{R})$ denote a conditional and marginal

PDF, respectively. Finally, let the Bayesian Mean Squared Error (MSE) [22, 47] between the clean speech magnitude $\underline{A}$ and the estimated speech magnitude $\hat{\underline{A}}$, be defined as

$$\mathcal{J}_{MSE} = \mathbb{E}_{\underline{A},\underline{R}} \left\{ \left(\underline{A} - \hat{\underline{A}} \right)^2 \right\}. \tag{18}$$

By minimizing the Bayesian MSE with respect to $\hat{\underline{A}}$ it can be shown (see e.g. [22, 47]) that the optimal STSA-MMSE estimator is given as

$$\hat{\underline{A}} = \mathbb{E}_{\underline{A}|\underline{R}}\{\underline{A}|\underline{R}\}, \tag{19}$$

which is nothing more than the expected value of the clean speech magnitude $\underline{A}$ given the observed noisy speech magnitude $\underline{R}$.

From Eq. (19) a large number of estimators can be derived by considering different distributions of $p(\underline{A}, \underline{R})$ [23]. For example, in the seminal work of Ephraim and Malah in [28], the STFT coefficients of the clean speech and noise were assumed to be statistically independent, zero-mean, Gaussian distributed random variables. This assumption is motivated by the fact that STFT coefficients become uncorrelated, and under a Gaussian assumption therefore independent, with increasing frame length. Based on these assumptions Eq. (19) simplifies [22, 28] to

$$\hat{A}(k) = G(\psi_k, \gamma_k) R(k), \tag{20}$$

where $G(\psi_k, \gamma_k)$ is a gain function that is applied to the noisy spectral magnitude $R(k)$, and

$$\psi_k = \frac{\mathbb{E}\{|A(k)|^2\}}{\mathbb{E}\{|V(k)|^2\}}, \tag{21}$$

and

$$\gamma_k = \frac{R^2(k)}{\mathbb{E}\{|V(k)|^2\}}. \tag{22}$$

The term ψ_k is referred to as *a priori* SNR, similarly to Eq. (14) since $\psi_k \approx {\xi_\omega}^3$, and γ_k is referred to as the *a posteriori SNR* as it reflects the SNR of the observed, or noise corrupted, speech signal. As seen from Eq. (20) the STSA-MMSE gain is a function of *a priori* and *a posteriori* SNR. However, although the Wiener gain in Eq. (13) is also a function of *a priori* SNR, the STSA-MMSE gain in general introduces less artifacts at low SNR than the Wiener gain, partially due to the *a posteriori* SNR [22, 48]. In fact, at high SNRs (SNR > 20

[3] Equality only holds if DTFT coefficients in Eq. (21) are computed for infinite sequences of stationary processes. Since they are DFT coefficients computed based on finite sequences, it follows that $\psi_k \approx \xi_\omega$.

dB) the gains from the Wiener filter and STSA-MMSE estimator converges to the same value [22, 28, 33].

Since the first STSA-MMSE estimator was proposed using a Gaussian assumption, a large range of estimators have been proposed with different statistical assumptions, and cost functions, in an attempt to improve the performance by utilizing either more accurate statistical assumptions, which are more in line with the true probability distribution of speech and noise, or cost functions more in line with human perception [23, 34–36, 38, 49–53]. Finally, note that similarly to the Wiener filters, the *a priori* SNR has to be estimated, e.g. using noise PSD tracking (see e.g. [23] and references therein), in order to use the STSA-MMSE estimators in practice.

1.1.3 Subspace Methods

The third class of enhancement algorithms are known as subspace-based algorithms, as they are derived primarily using principles from linear algebra and not, to the same degree, on principles from signal processing and estimation theory, as the previously discussed algorithms were [22]. The general underlying assumption behind these algorithms is that K-dimensional vectors of speech signals do not span the entire K-dimensional euclidean space, but instead are confined to a smaller M-dimensional subspace, i.e. $M < K$ [54, 55]. Specifically, let a stationary stochastic process representing a clean speech signal $\underline{X} = [X_1,\ X_2,\ \ldots,\ X_K]^T$ be defined as

$$\underline{X} = \sum_{m=1}^{M} C_m \underline{p}_m = \underline{\underline{P}}\,\underline{C}, \tag{23}$$

where C_m are zero-mean, potentially complex, random variables and $\underline{p}_m$ are K-dimensional linearly independent, potentially complex, basis vectors, e.g. complex sinusoids [54]. Here,

$$\underline{C} = [C_1,\ C_2,\ \ldots,\ C_M]^T \in \mathbb{R}^M, \tag{24}$$

and

$$\underline{\underline{P}} = \left[\underline{p}_1,\ \underline{p}_2,\ \ldots,\ \underline{p}_M\right] \in \mathbb{R}^{K\times M}, \tag{25}$$

and if $M = K$, the transformation between $\underline{X}$ and $\underline{C}$ is always possible as it corresponds to a change of coordinate system [54]. However, for speech signals, such a transformation is often possible for $M < K$ [54], which implies that $\underline{X}$ lies in a M-dimensional subspace spanned by the M columns of $\underline{\underline{P}}$ in the K-dimensional Euclidean space. This subspace, is commonly referred to as the signal subspace. Since the rank, denoted as $\mathcal{R}\{\cdot\}$, of $\underline{\underline{P}}$ is $\mathcal{R}\{\underline{\underline{P}}\} = M$, the covariance matrix of $\underline{X}$,

$$\underline{\underline{\Sigma}}_X = \mathbb{E}\{\underline{X}\underline{X}^T\} = \underline{\underline{P}}\,\underline{\underline{\Sigma}}_C\,\underline{\underline{P}}^T \in \mathbb{R}^{K\times K}, \tag{26}$$

where $\underline{\underline{\Sigma}}_C = \mathbb{E}\{\underline{C}\underline{C}^T\}$ is the covariance matrix of $\underline{C}$, will be rank deficient, $\mathcal{R}\{\underline{\underline{\Sigma}}_X\} = \mathcal{R}\{\underline{\underline{\Sigma}}_C\} = M < K$. Noting from the stationarity of $\underline{X}$ that $\underline{\underline{\Sigma}}_X \succeq 0$, it follows that $\underline{\underline{\Sigma}}_X$ only has non-zero eigenvalues. The fact that $\underline{\underline{\Sigma}}_X$ has some eigenvalues that are equal to zero is the key to subspace-based speech enhancement.

For convenience, let us rewrite our signal model from Eq. (1) in vector form,

$$\underline{Y} = \underline{X} + \underline{V}, \tag{27}$$

where $\underline{Y}$, $\underline{X}$, and $\underline{V}$ are the K-dimensional stochastic vectors representing the time-domain noisy speech signal, the clean speech signal, and noise signal, respectively. Employing the standard assumption that speech $\underline{X}$ and noise signals $\underline{V}$ are stationary, uncorrelated, and zero-mean random processes [28, 54] it follows that

$$\underline{\underline{\Sigma}}_Y = \underline{\underline{\Sigma}}_X + \underline{\underline{\Sigma}}_V, \tag{28}$$

where $\underline{\underline{\Sigma}}_Y$, and $\underline{\underline{\Sigma}}_V$ are the covariance matrices of the noisy speech signal, and noise signal, respectively. Furthermore, with the additional assumption that the noise signal is white, with variance σ_V^2, Eq. (28) reduces to

$$\underline{\underline{\Sigma}}_Y = \underline{\underline{\Sigma}}_X + \sigma_V^2 \underline{\underline{I}}_K, \tag{29}$$

where $\underline{\underline{I}}_K$ is the K-dimensional identity matrix. Now, consider the Eigen-Value Decomposition (EVD) of Eq. (29) given as

$$\underline{\underline{U}}\,\underline{\underline{\Lambda}}_Y \underline{\underline{U}}^T = \underline{\underline{U}}\,\underline{\underline{\Lambda}}_X \underline{\underline{U}}^T + \underline{\underline{U}}\,\underline{\underline{\Lambda}}_V \underline{\underline{U}}^T, \tag{30}$$

where $\underline{\underline{U}}$ is a matrix with the K orthonormal eigenvectors of $\underline{\underline{\Sigma}}_Y$, and $\Lambda = \text{diag}(\lambda_{y,1}, \lambda_{y,2}, \ldots, \lambda_{y,K})$ is a diagonal matrix with the corresponding K eigenvalues. Since it is assumed that $\mathcal{R}\{\underline{\underline{\Sigma}}_X\}$ is rank deficient (Eq. (23)) the eigenvalues of $\underline{\underline{\Sigma}}_Y$ can be partitioned in descending order based on their magnitude as

$$\lambda_{y,k} = \begin{cases} \lambda_{x,k} + \sigma_V^2 & \text{if } k = 1, 2, \ldots, M \\ \sigma_V^2, & \text{if } k = M+1, M+2, \ldots, K. \end{cases} \tag{31}$$

Then, it follows [22, 54] that the subspace spanned by the eigenvectors corresponding to the M largest eigenvalues of $\underline{\underline{\Sigma}}_Y$, i.e. the top line in Eq. (31), corresponds to the subspace spanned by the eigenvectors of $\underline{\underline{\Sigma}}_X$, which is the same subspace spanned by the columns of $\underline{\underline{P}}$, i.e. the signal subspace. Specifically, let, $\underline{\underline{U}}$ be partitioned as $\underline{\underline{U}} = [\underline{\underline{U}}_1 \; \underline{\underline{U}}_2]$ such that $\underline{\underline{U}}_1$ is a $K \times M$ matrix with the eigenvectors corresponding to the M largest eigenvalues of $\underline{\underline{\Sigma}}_Y$, and $\underline{\underline{U}}_2$ is a $K \times (K - M)$ with the remaining $K - M$ eigenvectors, then $\underline{\underline{U}}_1 \underline{\underline{U}}_1^T$ is a projection matrix that orthogonally projects its multiplicand onto the signal subspace. Similarly, $\underline{\underline{U}}_2 \underline{\underline{U}}_2^T$ will be the projection matrix that projects its multiplicand onto the complementary orthogonal subspace, known as the

noise subspace. Hence, it follows that a realization of the noisy signal can be decomposed as

$$\underline{y} = \underline{\underline{U}}_1 \underline{\underline{U}}_1^T \underline{y} + \underline{\underline{U}}_2 \underline{\underline{U}}_2^T \underline{y}. \tag{32}$$

Finally, since the noise subspace spanned by the columns of $\underline{\underline{U}}_2$ contains no components of the clean speech signal, the noise subspace can be nulled to arrive at an estimate of the clean speech signal given as

$$\hat{\underline{x}} = \underline{\underline{U}}_1 \underline{\underline{U}}_1^T \underline{y}. \tag{33}$$

In fact, the solution in Eq. (33) can, similarly to the previously discussed methods, be viewed as a gain-based approach (see Fig. 1) given by

$$\hat{\underline{x}} = \underline{\underline{U}} \underline{\underline{G}}_M \underline{\underline{U}}_1^T \underline{y}, \tag{34}$$

where $\underline{\underline{G}}_M$ is simply the M-dimensional identity matrix. In this form, a transformation $\underline{\underline{U}}_1^T \underline{y}$ is applied to the noisy time-domain speech signal $\underline{y}$, which in this case is the linear transformation matrix $\underline{\underline{U}}_1^T$, known as the Karhunen-Loève Transform (KLT). Then, a unit-gain $\underline{\underline{G}}_M$ is applied before an inverse KLT, $\underline{\underline{U}}$, is used to reconstruct the enhanced signal to the time-domain.

In fact, what differentiate most subspace-based speech enhancement methods is the choice of transform domain $\underline{\underline{U}}_1$ and the design of the gain matrix $\underline{\underline{G}}_M$. An alternative to the approach based on the EVD of the covariance matrix, is the Singular-Value Decomposition (SVD) of time-domain signals ordered in either Toeplitz or Hankel matrices [22]. Furthermore, the gain matrix can be designed with an explicitly defined trade-off between noise reduction and signal distortion and even to handle colored noise signals [22, 55–57].

Finally, what most subspace-based speech enhancement algorithms have in common is the need for estimating the covariance matrix of the clean speech, or noise, signal and the, generally time-varying, dimension of the signal subspace M. Naturally, if M is overestimated, some of the noise subspace is preserved, but if M is underestimated some of the signal subspace is discarded. Consequently, the quality of these estimates highly influences the performance of subspace-based speech enhancement algorithms. Nevertheless, it has been shown that these algorithms are capable of improving speech intelligibility for hearing impaired listeners wearing cochlear implants [58].

1.1.4 Machine Learning Methods

Common for all the previously discussed clean-speech estimators is that they are all, to some degree, derived using mathematical principles from probability theory, digital signal processing, or linear algebra. Consequently, they are based on various assumptions such as stationarity of the signals involved, uncorrelated clean-speech and noise signals, independence of speech and noise transform coefficients across time and frequency, etc. These assumptions are

all trade-offs. On one hand, they must reflect the properties of real speech and noise signals, while, at the other hand, they must be simple enough that they allow mathematical tractable solutions.

Furthermore, they all require information about some, generally unknown, quantity such as the noise magnitude $|v(k,m)|$ for spectral subtractive-based techniques, *a priori* SNR for the statistically optimal algorithms such as the Wiener filters or STSA-MMSE estimators, or the signal subspace dimension, or covariance matrices for the clean speech or noise signals, for the subspace-based techniques. These quantities need to be estimated, and their estimates are critical for the performance of the speech enhancement algorithm. Finally, although these techniques are capable of improving the quality of a noisy speech signal, when the underlying assumptions are reasonably met [48], they generally do not improve speech intelligibility for normal hearing listeners [59–67].

A different approach to the speech enhancement task, a completely different paradigm in fact, is to consider the speech enhancement task as a supervised learning problem [68]. In this paradigm, it is believed that the speech enhancement task can be learned from observations of representative data, such as a large number of corresponding pairs of clean and noisy speech signals.

Specifically, instead of designing a clean-speech estimator in closed-form using mathematical principles, statistical assumptions, and *a priori* knowledge, the estimator is defined by a parameterized mathematical model, that represents a large function space, potentially with universal approximation properties such as Gaussian Mixture Models (GMMs) [69], Artificial Neural Networks (ANNs) [70, 71], or Support Vector Machines (SVMs) [72, 73]. The parameters of these machine learning models are then found as the solution to an optimization problem with respect to an objective function evaluated on a representative dataset.

This approach is fundamentally different from the previously described techniques since no restrictions, e.g. about linearity, or explicit assumptions, e.g. about stationarity or uncorrelated signals, are imposed on the model. Instead, signal features which are relevant for solving the task at hand, e.g. retrieving a speech signal from a noisy observation, are implicitly learned during the supervised learning process. The potential big advantage of this approach is that less valid assumptions, made primarily for mathematical convenience, can be avoided and as we shall see in this section, and sections to come, such an approach might result in clean-speech estimators with a potential to exceed the performance of the non-machine learning based techniques proposed so far.

Basic Principles

The basic principle behind most machine learning based speech enhancement techniques can be formulated as

$$\underline{\hat{o}} = \mathcal{F}(\underline{h}(\underline{y}), \underline{\theta}), \tag{35}$$

where $\mathcal{F}(\cdot, \underline{\theta})$ denotes a parameterized model with parameters $\underline{\theta}$. The input signal $\underline{y}$ denotes the noisy speech signal and $\underline{h}(\cdot)$ is a vector-valued function that applies a feature transformation to the raw speech signal $\underline{y}$. The representation of the output $\underline{\hat{o}}$ depends on the application, but it could e.g. be the estimated clean-speech signal or the clean-speech STFT magnitude. The optimal parameters $\underline{\theta}^*$ are then found, without loss of generality, as the solution to the minimization problem given as

$$\underline{\theta}^* = \underset{\underline{\theta}}{\operatorname{argmin}} \quad \mathcal{J}(\mathcal{F}(\underline{h}(\underline{y}), \underline{\theta}), \underline{o}), \quad (\underline{y}, \underline{o}) \in \mathcal{D}_{train}, \tag{36}$$

where $\mathcal{J}(\cdot, \cdot)$ is a non-negative objective function, and $(\underline{y}, \underline{o})$ is an ordered pair, of noisy speech signals $\underline{y}$ and corresponding targets $\underline{o}$, e.g. clean-speech STFT magnitudes, from a training dataset $\mathcal{D}_{train}$. In principle, the optimal parameters $\underline{\theta}$ are given such that $\mathcal{J}(\mathcal{F}(\underline{h}(\underline{y}), \underline{\theta}^*), \underline{o}) = 0$, i.e. $\underline{\hat{o}} = \underline{o}$. However, as datasets are incomplete, model capacity is finite, and learning algorithms non-optimal, achieving $\mathcal{J}(\mathcal{F}(\underline{h}(\underline{y}), \underline{\theta}^*), \underline{o}) = 0$, might not be possible. In fact, it may not even be desirable as it may lead to a phenomena known as overfitting, where the model does not generalize, i.e. performs poorly, on data not experienced during training [68].

Instead, what one typically wants in practice is to find a set of near-optimal parameters $\underline{\theta}^\dagger$ that achieve a low objective function value on the training set $\mathcal{D}_{train}$, but also on an unknown test dataset $\mathcal{D}_{test}$, where $\mathcal{D}_{test} \not\subset \mathcal{D}_{train}$, i.e. $\mathcal{D}_{test}$ is not a subset of $\mathcal{D}_{train}$, but still assumed to share the same underlying statistical distribution. Such a model is likely to generalize better, which ultimately enable the use of the model for practical applications, where the data is generally unknown. In fact, overfitting is the Achilles' heel of machine learning, and controlling the amount of overfitting and acquiring good generalization, is key to successfully applying machine learning based speech enhancement techniques in real-life applications.

Machine Learning for Enhancement

Machine learning has been applied to speech enhancement for several decades [74–80], but until recently, not very successfully in terms of practical applicability. In one of the first machine learning based speech enhancement techniques [74] the authors proposed to use an ANN (ANNs are described in detail in Sec. 2) to learn a mapping directly from a frame of the noisy speech signal $\underline{y}_m$ to the corresponding clean speech frame $\underline{x}_m$ as

$$\underline{\hat{x}}_m = \mathcal{F}_{ANN}(\underline{y}_m, \underline{\theta}), \tag{37}$$

where $\mathcal{F}_{ANN}(\cdot,\cdot)$ represents an ANN. Although the technique proposed in [74] was trained on only 216 words and with a small network, according to today's standard, their proposed technique slightly outperformed a spectral subtractive-based speech enhancement technique in terms of speech quality, but not speech intelligibility. Furthermore, the ANN generalized poorly to speech and noise signals not part of the training set. Finally, it took three weeks to train the ANN on a, at the time, modern super computer, which simply made it practically impossible to conduct experimental research using larger ANNs with larger datasets. This might explain why little ANN based speech enhancement literature exists from that time, compared to the previously discussed methods, such as Wiener filters or STSA-MMSE estimators, which, in general, require far less computational resources.

Almost two decades later, promising results were reported in [78], where large improvements (more than 60%) in speech intelligibility was achieved using a speech enhancement technique based on GMMs. Specifically, they followed a gain-based approach (see Fig. 1), and estimated a Time-Frequency (T-F) gain $\hat{g}(k,m)$ for each frequency bin k and time-frame m. The frequency decomposition of the time-domain speech signal was performed using a Gammatone filter bank with 25 channels [81] and the gain was defined as

$$\hat{g}^{IBM}(k,m) = \begin{cases} 1 & \text{if } \mathcal{P}(\pi_1|r(k,m)) > \mathcal{P}(\pi_0|r(k,m)) \\ 0 & \text{otherwise,} \end{cases} \tag{38}$$

where $\mathcal{P}(\pi_0|r(k,m))$ and $\mathcal{P}(\pi_1|r(k,m))$ denote the probabilities of the clean speech magnitude $|x(k,m)|$ belonging to one out of two classes. The two classes π_0, and π_1, denoted noise-dominated T-F units and speech-dominated T-F units, respectively, and were defined as

$$r(k,m) \in \begin{cases} \pi_1 & \text{if } \frac{|x(k,m)|^2}{|v(k,m)|^2} > T_{SNR}(k) \\ \pi_0 & \text{otherwise,} \end{cases} \tag{39}$$

where $\frac{|x(k,m)|^2}{|v(k,m)|^2}$ is the SNR in frequency bin k and time frame m and $T_{SNR}(k)$ is an appropriately set frequency-dependent threshold. The probabilities $\mathcal{P}(\pi_0|r(k,m))$ and $\mathcal{P}(\pi_1|r(k,m))$ were estimated using two classifiers, one for each class, based on 256-mixture GMMs[4] trained on 390 spoken utterances ($\approx$ 16 min of speech) with a feature representation based on Amplitude Modulation Spectrogram (AMS) [82]. In fact, the binary gain defined by Eq. (38) is an estimate of the Ideal Binary Mask (IBM), which is simply defined by Eqs. (38) and (39) when oracle information about $|x(k,m)|^2$ and $|v(k,m)|^2$ is used. Furthermore, it has been shown that the IBM can significantly improve intelligibility of noisy speech, even at very low SNRs [83–85], which makes

[4]Interestingly, in retrospect, they did attempt to use ANNs, but without good results.

the IBM a highly desirable training target as speech intelligibility is likely to be increased if the mask is accurately estimated.

This approach, first proposed in [78], was reported to not only outperform classical methods such as the Wiener filter and STSA-MMSE estimator, it even achieved improvements in speech intelligibility at a scale not previously observed in the speech enhancement literature. Later, supporting results appeared in [79, 86] where even better performance was achieved using a binary classifier based on SVMs.

However, it was later discovered [87] that the great performance achieved by the systems proposed in [78, 79] was primarily due to the reuse of the noise signal in both the training data and test data. This meant the systems in [78, 79] were tested on realizations of the noise signal that were already used for training. In theory, it allowed the models to "memorize" the noise signal and simply subtract it from the noisy speech signal during test. This, obviously is not a possibility in real-life applications, where the exact noise signal-realization is generally not known in isolation.

Regardless of the unrealistically good performance of the systems in [78, 79] they, combined with the co-occurring Deep Learning revolution (described in detail in Sec. 2), reignited the interest in machine learning based speech enhancement.

1.2 Classical Speech Separation Algorithms

We now extend the formulation of the classical speech enhancement task (see Eq. (1)) to multi-talker speech separation. Let $x_s[n]$ be a sample of a clean time-domain speech signal from speaker s, and let an observation of a mixture $y[n]$ be defined as

$$y[n] = \sum_{s=1}^{S} x_s[n], \tag{40}$$

where S is the total number of speakers in the mixture. Then, the goal of single-microphone multi-talker speech separation is to acquire estimates $\hat{x}_s[n]$ of $x_s[n]$, $s = 1, 2, \ldots, S$, which in some sense are "close to" $x_s[n]$, $s = 1, 2, \ldots, S$ using $y[n]$ only. In Sec. 1 we have seen a large number of techniques proposed to solve the single-microphone speech enhancement task, and to some extent, they are fairly successful in doing so in practice. However, they all, except for the machine learning based techniques, rely heavily on specific statistical assumptions about the speech and noise signals. Specifically, in practice, the Wiener filters and STSA-MMSE estimators rely on accurate estimates of the noise PSD.

Similarly, the subspace based techniques assume the noise signal is statistically white, or can be whitened, which in general requires additional

information about the noise signal. Consequently, if the noise signal is a speech signal, which is non-stationary and colored, the methods described in Sec. 1 perform poorly, as they rely on a signal model whose parameters are hard to estimate accurately. This, in turn, might also explain why these techniques have not been successfully applied on the single-microphone multi-talker speech separation task given by Eq. (40), as different techniques might be required to successfully handle speech-like interference signals, when the target signals themselves are speech. In this section we introduce some of the classical techniques, i.e. non-deep learning-based methods, that have been proposed for single-microphone multi-talker speech separation.

1.2.1 Harmonic-Models

Some of the early techniques for single-microphone speech separation were in fact more related to speech enhancement, than speech separation, as they aimed mainly at suppressing an interfering speaker, in a two-speaker mixture, than actually separating the speech signals (see e.g. [88–91]). Furthermore, compared to the speech enhancement techniques existing at the time, e.g. Wiener filters or STSA-MMSE estimators, the techniques proposed for speech separation were more involved compared to the simple gain-based approach used by the corresponding speech enhancement techniques. Finally, although more complicated, they were generally less successful as good performance could only be achieved by using *a priori* knowledge generally not available in practice, such as information about the fundamental frequency of the interfering speaker [89–91, 91]. For example, the techniques proposed in [88–90, 90] used the fact that voiced speech signals can be modeled by a sum of sinusoids and that two competing speakers generally have different fundamental frequency and consequently different harmonics. If knowledge about the fundamental frequency of the interfering speaker is known, one can, in theory, null that frequency and corresponding harmonics and suppress the interfering speaker. However, such an approach requires multi-pitch tracking as the individual pitch signals must be estimated from the noisy signal only, and such an approach only works if the fundamental frequencies of the speakers involved are sufficiently separated. Finally, as unvoiced speech does not possess any apparent harmonic structure [92], these techniques were only partly successful and not a general approach for single-microphone multi-talker speech separation [88–90, 90].

1.2.2 Computational Auditory Scene Analysis

A different approach to the single-microphone multi-talker speech separation task is one based on principles from Auditory Scene Analysis (ASA) [9]. According to the ASA paradigm, the auditory system works by decomposing acoustic signals into abstract objects known as auditory streams. For

each acoustic source signal that impinge upon the eardrum, e.g. speech or environmental sounds, an auditory stream is produced, which enables the conscious or subconscious mind to focus on every one of them in isolation. These auditory streams are thought to be produced in two steps [9]. First, a segmentation step decomposes the acoustic time-domain signal into individual units in a T-F domain representation, where it is assumed that each T-F unit primarily originates from a single source. Secondly, these T-F units are grouped into auditory streams based on grouping rules known as sequential grouping or simultaneous grouping. Sequential grouping assumes T-F units that are similar across time belong to the same source and consequently are grouped into the same auditory stream, whereas simultaneous grouping merge T-F units that share similarities across frequency, e.g. harmonicity or common onsets and offsets. Designing algorithms to separate speech signals based on the principles of ASA, is referred to as Computational Auditory Scene Analysis (CASA) [12, 93] and multiple techniques have been proposed to solve the single-microphone multi-talker speech separation task (see e.g. [93–99]).

For example, in [94] a CASA system is proposed for co-channel separation of voiced speech using grouping rules based on cross-correlation analysis across cochlear channels (simultaneous grouping) and modulation analysis of cochlear-filter responses across time (temporal grouping). From these grouping rules, a binary mask is generated that is applied to the T-F representation of the mixture signal to separate the target speaker from the interference signal. In [98] a system is proposed for co-channel speech separation of both voiced and unvoiced speech. Simultaneous grouping of the voiced parts of the co-channel signal is performed using an iterative pitch tracking algorithm that identifies the dominant fundamental frequency in the mixture. Sequential grouping is then formulated as a clustering problem that uses the information from the simultaneous grouping step and cluster T-F units across time that belong to the same speaker. Using information about the voiced segments of the co-channel signal the unvoiced parts of the mixture signal are identified using onset/offset analysis and a binary mask for the entire co-channel speech signal is computed and used to separate the speech signals.

Although CASA based single-microphone multi-talker speech separation algorithms can be somewhat successful, they suffer from several drawbacks. For example, CASA is to a large extent based on heuristics and manually-designed grouping and segmentation rules, which might not be optimal. Furthermore, these rules are primarily based on speech characteristics and, consequently, are not valid for non-speech signals. Finally, these systems typically lack the capability to learn from data.

1.2.3 Non-Negative Matrix Factorization

An alternative to the primarily heuristically based CASA approaches are the more mathematically founded Non-negative Matrix Factorization (NMF) based algorithms [100]. The underlying assumption behind these algorithms is that a non-negative data matrix $\underline{\underline{V}} \in \mathbb{R}^{K \times M}$ can be approximately factorized as

$$\underline{\underline{V}} \approx \underline{\underline{D}}\,\underline{\underline{H}}, \tag{41}$$

where $\underline{\underline{D}} \in \mathbb{R}^{K \times L}$, $\underline{\underline{H}} \in \mathbb{R}^{L \times M}$, and $L \ll M$ are non-negative matrices representing a dictionary matrix and an activation matrix, respectively. The dictionary matrix $\underline{\underline{D}}$ can be seen as a set of basis vectors for the data matrix $\underline{\underline{V}}$ and the columns of the activation matrix $\underline{\underline{H}}$ represent how much of each basis vector is needed to represent each column of $\underline{\underline{V}}$. The dimension L is a tuning parameter that controls the accuracy of the approximation and is specified experimentally. Obviously, if $L = M$, the solution $\underline{\underline{V}} = \underline{\underline{D}}\,\underline{\underline{H}}$ can always be found, although such a decomposition is of no interest as no compact representation of $\underline{\underline{V}}$ is found. Factorizing $\underline{\underline{V}}$ into $\underline{\underline{D}}$ and $\underline{\underline{H}}$, where $L \ll M$, can be achieved in an iterative fashion [101] by updating

$$\underline{\underline{H}} = \underline{\underline{H}} \circ \frac{\underline{\underline{D}}^T \underline{\underline{V}}}{\underline{\underline{D}}^T \underline{\underline{D}}\,\underline{\underline{H}}}, \tag{42}$$

and

$$\underline{\underline{D}} = \underline{\underline{D}} \circ \frac{\underline{\underline{V}}\,\underline{\underline{H}}^T}{\underline{\underline{D}}\,\underline{\underline{H}}\,\underline{\underline{H}}^T}, \tag{43}$$

where $\circ$ and $-$ denote element-wise multiplication and division, respectively. In fact, Eqs. (42) and (43) represent the solution to the least squares optimization problem given by

$$\begin{aligned} \underset{\underline{\underline{D}}, \underline{\underline{H}}}{\text{minimize}} \quad & \|\underline{\underline{V}} - \underline{\underline{D}}\,\underline{\underline{H}}\|_F^2 \\ \text{subject to} \quad & \underline{\underline{D}}, \underline{\underline{H}} \geq 0, \end{aligned} \tag{44}$$

where $\|\cdot\|_F^2$ is the squared Frobenius norm, and by iterating between Eqs. (42) and (43), convergence to the optimal solution of Eq. (44) is guaranteed [101].

NMF is a data-driven technique and signal-dependent dictionaries $\underline{\underline{D}}$ are generally required [100]. In the speaker separation case, a dictionary $\underline{\underline{D}}_s$, $s = 1, 2, \ldots, S$ is typically used for each speaker (see e.g. [102–106]) and noise source if any is present [107]. Let $r(k, m)$ and $a_s(k, m)$ denote STFT magnitudes for time-frame m and frequency bin k for the noisy speech signal $y[n]$ and clean speech signal $x_s[n]$, from speaker s, respectively. Furthermore, let $\underline{\underline{R}} \in \mathbb{R}^{K \times M}$ and $\underline{\underline{A}}_s \in \mathbb{R}^{K \times M}$ denote spectrogram matrices populated with $r(k, m)$ and $a_s(k, m)$, for suitable ranges of variables k, m, respectively.

From a matrix $\underline{\underline{A}}_s$, a speaker specific dictionary $\underline{\underline{D}}_s$, for speaker s, can be obtained using Eqs. (42) and (43) with $\underline{\underline{V}} = \underline{\underline{A}}_s$. To acquire an estimate $\hat{\underline{\underline{A}}}_s$ from a mixture signal $\underline{\underline{R}}$ containing unknown realizations of multiple speakers, and noise sources, Eq. (42) is used with Eq. (43) being fixed, and with $\underline{\underline{V}} = \underline{\underline{R}}$ and $\underline{\underline{D}} = \underline{\underline{D}}_s$. When Eq. (42) has converged, the corresponding activation matrix $\underline{\underline{H}}_s$ is used to acquire and estimate of $\underline{\underline{A}}_s$ as

$$\hat{\underline{\underline{A}}}_s = \underline{\underline{D}}_s \underline{\underline{H}}_s \quad s = 1, 2, \ldots, S. \tag{45}$$

Finally, using the phase of the mixture signal $\underline{\underline{R}}$, the overlap-add technique [25], and the Inverse Discrete Fourier Transform (IDFT), the time-domain signals $\hat{x}_s[n]$, $s = 1, 2, \ldots, S$, are obtained for each separated speaker.

NMF is a simple but powerful technique for single-microphone multi-talker speech separation, or speech enhancement for that matter [108–110]. However, NMF has multiple drawbacks. First, NMF is a linear model and as such is limited in its model capacity. Second, NMF generally requires speaker dependent dictionaries, making NMF less suitable for speaker, or noise-type independent applications. Third, as signal-dependent activations $\underline{\underline{H}}_s$ are required in Eq. (45), it is not straight forward to apply NMF for real-time applications where low latency is critical. Finally, due to the computational complexity of the update equations (Eqs. (42) and (43)), NMF does not scale well to large datasets.

1.2.4 Generative Models

It is well known that speech signals are highly structured with temporal dynamics on multiple levels [2, 4, 21]. For example, at the phone level, i.e. the physical speech sounds, structure exist due to e.g. prosody or physiological variations among humans (e.g. differences in fundamental, and formant frequencies), but also at the phoneme level, speech is structured due to e.g. grammar and language, but also due to phenomena like co-articulation.

The speech separation algorithms discussed so far, such as CASA or harmonic-model based techniques, consider this temporal structure only partially, while NMF based algorithms do not take it into account at all. These drawbacks, of existing methods not fully utilizing the available information in speech signals, has motivated research in a different class of algorithms based on Hidden Markov Models (HMMs). Differently from the previously discussed methods, HMMs are generative stochastic models, and they have an internal discrete state representation that allow them to learn temporal dynamics of sequential data [68]. Specifically, a HMM is a finite state machine that changes among a discrete number of states in a synchronous manner. For each time step, the state of the HMMs, which is a latent variable, changes from one state to another based on a set of transition probabilities. Associated with each state is a set of observed stochastic variables, known as emission

probabilities, which are typically represented as a GMM. The parameters of a HMM, i.e. transition and emission probabilities, are typically found such that they maximize a certain likelihood function with respect to a given dataset, and although a HMM is a generative model, it is generally not used as such for speech processing tasks. For example, for Automatic Speech Recognition (ASR), phoneme-specific HMMs can be designed to model the distribution over speech signals containing certain phonemes. Then, to recognize an unknown phoneme from a speech signal, the conditional probability of the observed data, given a HMM, is evaluated for each of the phoneme-specific HMMs and the phoneme associated with the HMM of largest conditional probability, will be assigned as the phoneme in the speech signal [21].

If, however, more than one signal of interest is present in the speech signal, such as in the multi-talker speech separation task, the standard HMM framework can be extended into what is known as factorial HMMs [111]. These factorial HMM allow for more than one latent variable, hence allowing for generative models that can model speech mixtures containing multiple simultaneous sources, which ultimately led to the development of a large number of successful algorithms for single-microphone multi-talker speech separation [34, 99, 112–120]. In fact, the use of factorial HMM led to a major milestone in speech separation research, as a single-microphone two-talker speech separation system was shown to be capable, in a narrow setting, to separate two-talker speech such that a machine, i.e. an ASR system, could transcribe the speech signal better than humans [118, 119].

However, although factorial HMMs showed impressive results in [118, 119], they do have some drawbacks. For example, computing the conditional probabilities required during training and test, is intractable [117] and consequently these probabilities have to be estimated, which in general has a high computational complexity and scales poorly with the number of speakers [120]. Also, as factorial HMMs for multi-talker speech separation require speaker-dependent HMMs for each speaker in the mixture, these techniques can only be applied in a speaker dependent context, where the identities of the speakers to be separated are known *a priori*. This is a limitation that makes the factorial HMM framework non-applicable in a range of real-world applications such as automatic meeting transcription or automatic captioning for audio/video recordings, where the identity and number of speakers are generally unknown. These limitations also explain why the techniques proposed in [34, 112–120] primarily considered two-talker speech separation of a limited number of *known* speakers. Consequently, different techniques are required to enable multi-talker speech separation algorithms to work in such general applications, where only a limited amount of *a priori* knowledge about the environment is available. Potential candidates that might work in such environments are algorithms based on deep neural networks, which will be presented in detail in Sec. 3.2.

1.3 Evaluation

As mentioned in Sec. 1, the common goal of many speech enhancement, and multi-talker speech separation, algorithms is to improve either speech quality or intelligibility, of a degraded speech signal. But how do you accurately evaluate if an algorithm-under-test really does improve one of these quantities?

In general, the only way to truly evaluate if a speech processing algorithm in fact does improve speech quality or intelligibility, is by a listening test involving the end user, i.e. human test subjects. However, listening tests are involved as they require numerous human test subjects and the listening test itself, needs to be carefully planned based on whether the goal is to evaluate speech intelligibility or speech quality. Most people probably have an idea about what a good quality speech signal sounds like, and what would make the same signal a bad quality one, e.g. by introducing hiss or crackle sounds to the signal. Nevertheless, speech quality is highly subjective as it is primarily based on emotions and feelings. Speech intelligibility, on the other hand, is much more objective, if you will, as emotions and feelings in general do not influence your capability of understanding speech. Either you understand what is being said or you do not. Consequently, designing listening tests that truly evaluate speech quality or intelligibility, is no easy task [21, 22].

Therefore, to avoid these often tedious and time consuming listening tests, and to get a quick and somewhat accurate estimate of the listening-test result, a set of objective measures have been designed, which are based on mathematical functions that quantify the difference between clean and noisy/processed speech signals in a way that has a high correlation with listening-test results. In fact, in some cases, it is more desirable to use an objective measure, instead of a listening test involving human test subjects, as objective measures are fast, cheap, and consistently produce the same result for the same testing condition, whereas listening-test results might vary due to factors such as listener fatigue, or varying hearing ability among test subjects.

In the following, three of the popular techniques for objective quality and intelligibility evaluation are briefly reviewed.

1.3.1 Perceptual Evaluation of Speech Quality

The Perceptual Evaluation of Speech Quality (PESQ) [121–124] measure is one of the most widely used objective measures for estimating speech quality [22]. The PESQ measure is designed to approximate the Mean Opinion Score (MOS), which is a widely used listening test procedure for speech quality evaluation [21, 22, 122, 125]. The MOS is a very simple evaluation procedure, where the test subjects are asked to grade the speech signal they

are hearing based on a scale with five discrete steps, with "1" representing a bad and very annoying sound quality, and "5" representing an excellent sound quality with imperceptible distortions. The final MOS score, which is a single scalar between "1" and "5", is simply the average, or mean, of all the "opinion scores" for each test signal and for all test subjects, hence the name, mean opinion score. As mentioned, the PESQ measure approximates MOS, but the PESQ algorithm is fairly complex as it consists of multiple steps involving pre-processing, time alignment, perceptual filtering, masking effects, etc. (see e.g. [22, pp. 491-503]). Nevertheless, PESQ versions P.862.1/2 [123, 124] produce a number ranging from approximately 1 to 4.5, which allow comparisons between PESQ and MOS, and PESQ has been found to be highly correlated with listening-test experiments based on MOS [121, 123]. In fact, although PESQ was originally designed for evaluating speech coding algorithms, it was later shown that PESQ correlated reasonably well with the quality of speech processed by commonly used speech enhancement algorithms [126]. Also, PESQ requires both the clean speech signal as well as the noisy/processed signal to estimate the perceived quality of the noisy/processed signal. This makes PESQ an intrusive speech quality estimator, which limits its use to situations where the clean undistorted signal is available in isolation. For most applications of PESQ, this is not a real limitation as PESQ is usually used in laboratory conditions, where the clean signal is often available in isolation.

1.3.2 Short-Time Objective Intelligibility

The Short-Time Objective Intelligibility (STOI) [127, 128] is, today, perhaps, the most widely used objective measure for estimating speech intelligibility. Differently from PESQ, STOI is not designed to approximate any specific type of listening test, but merely designed to correlate well with listening test evaluating speech intelligibility in general. Since intelligibility is binary in the sense that, either a given speech signal, say a word, is understood or it is not, listening-test results representing speech intelligibility can, most often, be quantified as a number between 0 and 1 that represents the percentage of words correctly understood [22]. To be comparable with such tests, STOI is designed to produce a single scalar output in a similar range[5], with an output of 1 indicating fully intelligible speech.

Similarly to PESQ, STOI is an intrusive algorithm as it requires both the clean signal and the noise/processed signal in isolation. Furthermore, STOI is based on the assumption that modulation frequencies play an important role in speech intelligibility, and that all frequency bands in the cochlear filter are equally important. These are assumptions, which, to a certain degree,

[5] In theory, STOI can produce numbers in the interval $(-1, 1)$, since STOI is based on a correlation coefficient measure. However, in practice, negative numbers are rarely observed.

are justified empirically [4, 129, 130]. This also has the consequence that, compared to PESQ, STOI is a fairly simple algorithm.

Despite its simple formulation, STOI has been found to be able to quite accurately predict the intelligibility of noisy/processed speech in a wide range of acoustic scenarios [128, 131–134]. Finally, an extension to STOI, known as Extended Short-Time Objective Intelligibility (ESTOI), has been proposed as a more accurate speech intelligibility predictor in the special cases where the noise sources are highly modulated [135].

1.3.3 Blind Source Separation Evaluation

When evaluating single-target signal speech processing algorithms, such as speech enhancement, PESQ and STOI are useful, as these measures quantify how successful the algorithm-under-test process a degraded signal in a way that is perceptually desirable. If, however, multiple target signals exist, such as in a speech separation task, additional information about the processing artifacts might be desirable compared to what PESQ and STOI can provide [136, 137]. In other words, when a mixture signal that contains multiple speech and noise signals are processed by a speech separation algorithm, the enhanced or separated speakers might contain artifacts originating from multiple different sources. For example, these artifacts could originate from the noise signal itself, from processing artifacts, or due to "cross-talk", i.e. signal components from one target speaker appearing in the separated signal of the other.

One of the most popular objective measures for evaluating speech separation algorithms that take these considerations into account, is the Blind-Source Separation (BSS) Eval toolkit [138]. In the technique proposed in [138], the separated signals are decomposed into target-speaker components and three noise components known as interference, noise, and artifact. The interference component represents cross-talk from other target speakers. Noise and artifacts, represents environmental noise sources and processing artifacts, respectively. From this decomposition, energy-ratio measures are defined known as Source-to-Distortion Ratio (SDR), Source-to-Interference Ratio (SIR), Source-to-Artifact Ratio (SAR), and SNR, which each relate these decomposed elements of the separated signal in a way that provide useful information about the contribution of each of them. Finally, it has been found that these objective measures correlate well with listening test evaluating quality [139, 140], and, obviously, the BSS Eval toolkit only compliments other objective measures such as STOI and PESQ.

2 Deep Learning

Section 1 reviewed classical techniques that have been proposed to solve the single-microphone speech enhancement and single-microphone multi-talker speech separation tasks. Although, these techniques are very different, and try to solve different tasks, most of them rely heavily on one key component: *domain knowledge*. For example, the Wiener filters and STSA-MMSE estimators reviewed in Sec. 1.1 are designed based on the assumption that speech and noise have different statistical characteristics that are governed by basic probability distributions. Similarly, techniques reviewed in Sec. 1.2 for multi-talker speech separation, such as the CASA-based techniques, rely on detailed knowledge about the human auditory and speech production systems. Obviously, domain knowledge is extremely helpful when it is correct, and it has, and still is, used to solve many engineering problems. However, for complex tasks, utilizing domain knowledge might not be easy and it might even be destructive if the wrong assumptions are used. In other words, domain knowledge is useful for tasks that are well understood by the human engineers working on them. Instead, for complex tasks, it might lead to more successful solutions if the solution is learned, e.g. using a reinforcement strategy similarly to what is used in nature [141]. This philosophy, *learning* the solution instead of designing it, is one of the defining principles in the deep learning paradigm [20, 142, 143] and combined with the other defining principle, *depth*, it led to the deep learning revolution we know today, which is the topic of this section.

2.1 The Deep Learning Revolution

We are currently experiencing a deep learning revolution [20, 144], and although *deep learning* as an everyday term is less than a decade old, some of the fundamental principles used by deep learning algorithms today, dates back more than half a century [145]. In fact, the first successful application of the prototypical learning model used in deep learning, the Artificial Neural Network (ANN) (to be introduced in Sec. 2.2), was achieved by Frank Rosenblatt in the late 1950's with his *perceptron* learning model [146, 147], and shortly after by Bernard Widrow with a model known as ADALINE [148]. These models, heavily inspired by psychology and neuroscience [149, 150], were designed in an attempt to model the neural networks comprising the human brain, hence the name ANN.

However, although these models were capable of solving simple pattern recognition tasks, a decade later it was proven by Marvin Minsky and Seymour Papert [151, 152] that perceptrons were, in fact, inherently limited due to their linear input-output mapping and could indeed solve only very simple tasks. Interestingly, it was already known at the time that a simple way

to alleviate the limitations of the perceptron was to change the linear mapping to a more complex non-linear mapping by simply stacking multiple perceptrons into models known as Multi-Layer Perceptrons (MLPs). These more advanced, and deeper, ANN models were much more capable in terms of representational capacity, and could potentially solve extremely complex problems [145, 151, 152]. Unfortunately, at the time, training these MLPs was not feasible, partly due to lack of sufficient computational resources, but primarily due to lack of successful training algorithms.

It took almost two decades before an efficient algorithm for training MLPs were developed, which was popularized as *back-propagation* by Rumelhart *et al.* in 1986 [153][6]. Not long after the invention of back-propagation for training MLPs, theoretical results were published, proving that MLPs can approximate practically any function, with any desired accuracy. These theoretical results are known as the *universal approximation theorem* for MLPs [70, 71, 155]. The results were encouraging as they settled some of the speculations about the lack of potential of MLPs put forward by Minsky and Papert [151, 152] and proved, once and for all, that MLPs indeed did have the potential to solve complex tasks.

With the awareness of the back-propagation algorithm [153] and the universal approximation theorem [70, 71, 155], a natural question arises: why did it take two decades from the mid 1980s to the mid 2000s, before MLPs, or ANNs in general, became popular and practically applicable? The short answer to this question is: due to lack of labeled data and computational resources [156–158]. It was, however, a technique known as *unsupervised pre-training* that ignited the deep learning revolution in 2006 with two seminal papers by Hinton *et al.* [159, 160].

At the time, it was a general misconception that the optimization of MLPs, or Deep Neural Networks (DNNs) as they are usually called today, got trapped in poor local minima, hence preventing DNNs with multiple non-linear layers to be efficiently trained [161]. In an attempt to alleviate this presumed challenge it was proposed to initialize the parameters of the DNNs, before back-propagation training, with the parameters of a generative model, known as a Deep Belief Network (DBN) [160, 161]. The intuition behind this was that if you consider two random variables X and Y, and wish to learn $P(Y|X)$ it might be useful to first learn $P(X)$ using a generative model. DBNs are generative models constructed by stacking multiple Restricted Boltzmann Machines (RBMs), and then trained unsupervised using unlabeled data to model $P(X)$. RBMs are themselves generative models and belong to a broader class of undirected probabilistic graphical models, known as Markov Random Fields (MRFs) [68]. Inference in MRFs, however,

[6] Although Rumelhart *et al.* [153] coined the term back-propagation, some argue (see e.g. [152, 154]) that they did not invent the algorithm, they simply popularized it.

is challenging as it requires the evaluation of a, generally intractable, partition function. However, Hinton *et al.* showed in [160] that RBMs can be combined into a DBN and trained efficiently in a greedy layer-wise fashion using an approximate inference algorithm known as contrastive divergence [162]. The parameters of this DBN, which is modeling $P(X)$ can then be used to initialize a DNN, which is then "fine-tuned" using the traditional supervised back-propagation technique to model $P(Y|X)$.

In the seminal paper [160] Hinton *et al.* showed that DNNs initialized with unsupervised pre-training and refined with supervised back-propagation training, could achieve state-of-the-art results on a hand-written digits recognition task. The results attracted a huge amount of attention from the academic community and ultimately sparked the renewed interest in DNNs. It was, however, later recognized (see e.g. [20, 158, 163–168]) that poor local minima in general was not a problem when training DNNs and similar or even better performance could be achieved without using unsupervised pre-training, especially for large labeled dataset. Consequently, today, unsupervised pre-training is a technique that is rarely used, but its influence and impact on the scientific field of DNNs, as a catalyst for DNN research, cannot be overstated.

Over the last decade, deep learning has truly revolutionized both academia and industry. For example, deep learning technology has facilitated the development of algorithms that are close to, or even exceeding, human-level performance within multiple scientific disciplines such as automatic speech recognition [169–172], object recognition [173], face recognition [174], lip reading [175], board and computer games [176–179], and in healthcare applications [180, 181] especially for cancer detection [182–186].

Furthermore, today, deep learning is the key technology of many companies, and although the deep learning revolution was initiated by Hinton *et al.*, it was, in fact, the increase in low-cost computational resources made available by the general-purpose graphics processing unit [158, 187, 188] that really facilitated the success of deep learning and allowed DNNs to be applied on an industrial scale. For example, Facebook currently uses DNNs to predict and analyze user behavior 200 trillion, i.e. 200×10^{12}, times each day, something that was practically impossible just a decade ago [189, 190].

Finally, in a recent study [191] by PricewaterhouseCoopers, it is estimated that deep learning will contribute $15.7 trillion to the global economy in 2030, which is more than the current output of China and India combined. These contributions will be within a wide range of sectors such as health care, automotive, financial services, transportation, logistics, retail, energy, and manufacturing, which also justifies why deep learning driven technology is believed to lead to *the fourth industrial revolution* [192, 193].

2.2 Feed-Forward Neural Networks

Sections 2 and 2.1 introduced deep learning without defining exactly what a DNN is. In this section, and sections to come, we will introduce three of the most popular DNN models: Feed-forward Neural Networks (FNNs) (Sec. 2.2), Recurrent Neural Networks (RNNs) (Sec. 2.3), and Convolutional Neural Networks (CNNs) (Sec. 2.4) [158].

A FNN is a machine learning model and is represented as a parameterized function given by

$$\underline{\hat{o}} = f(\underline{y}, \theta), \tag{46}$$

where $\underline{y}$ is an input vector, θ is a set of parameters and $\underline{\hat{o}}$ is the FNN output, i.e. the map of $\underline{y}$ by $f(\cdot, \cdot)$. The most basic FNN is a single-layer FNN given by

$$\underline{\hat{o}} = f^{\{1\}}(\underline{y}, \theta) = \phi(\underline{\underline{W}}\underline{y} + \underline{b}), \quad \theta = \{\underline{\underline{W}}, \underline{b}\}, \tag{47}$$

where $\underline{\underline{W}}$ and $\underline{b}$ are the parameters and $\phi(\cdot)$ is a, generally non-linear, function known as the activation function. The vector $\underline{b}$ is known as the bias vector and allows the FNN to apply an affine transformation to $\underline{y}$. The prefix *feed-forward* in FNNs comes from the fact that information only flows in one direction in the model in Eq. (47), i.e. there are no recurrent connections. Furthermore, if $\phi(\cdot)$ is a binary thresholding function, Eq. (47) resembles the perceptron [146] or ADALINE models [148] and as shown by Minsky and Papert [151], these models are inherently limited as the input to the binary thresholding function is a purely linear transformation of the input $\underline{y}$. Instead, if these models are stacked as

$$\underline{\hat{o}} = f^{\{L\}}(\dots f^{\{2\}}(f^{\{1\}}(\underline{y}, \theta_1), \theta_2) \dots, \theta_L) \tag{48}$$

they form MLPs or multi-layer FNNs, which according to the universal approximation theorems (see e.g. [70, 71, 155]) can model practically any function. In fact, $L = 2$ is sufficient for the universal approximation theorem to apply, although it might require an exponentially wide network, i.e. number of rows in $\underline{\underline{W}}$, to approximate a certain function with a given accuracy. However, as the number of layers L increases, the compositional structure allow multi-layer FNNs to construct exponentially more complex decision boundaries, with the same number of parameters, than FNNs with $L = 2$ [194–196]. The fact that FNNs gets exponentially more efficient, with respect to the parameters, as the number of layers increase, is exactly what drives the deep learning research community for increasingly deeper networks, and although deep networks are not trivial to train, modern deep learning models have been trained successfully with more than 1000 layers [197, 198].

Similarly to the machine learning methods presented in Sec. 1.1.4, the optimal parameters of a FNN are typically given as the solution to an optimiza-

tion problem on the form

$$\theta^* = \underset{\theta}{\operatorname{argmin}} \sum_{\mathcal{D}_{train}} \mathcal{J}(f^{\{L\}}(\underline{y}, \theta), \underline{o}), \quad (\underline{y}, \underline{o}) \in \mathcal{D}_{train}, \tag{49}$$

where $\mathcal{J}(\cdot, \cdot) \in \mathbb{R}$ is a non-negative cost function (e.g. mean squared error), $f^{\{L\}}(\underline{y}, \theta)$ represents a FNN with L layers, $\theta = \{\theta_1, \theta_2, \ldots, \theta_L\}$ is the set of parameters, and $(\underline{y}, \underline{o})$ is an ordered pair, of input signals $\underline{y}$ and corresponding targets $\underline{o}$, from a training dataset $\mathcal{D}_{train}$. The objective is then to find parameters θ such that $\underline{\hat{o}} = f^{\{L\}}(\underline{y}, \theta) \approx \underline{o}$. However, until 1986 a solution to Eq. (49) was only known for $L = 1$, but when Rumelhart *et al.* [153] proposed the back-propagation algorithm, Eq. (49) could be solved, in theory, for any L using the optimization method gradient descent (e.g. [199–201]).

Specifically, Rumelhart *et al.* proposed to update any weight w in a multi-layer FNN using gradient descent defined as

$$w^{(n+1)} = w^{(n)} - \mu \frac{\partial \mathcal{J}}{\partial w^{(n)}}, \tag{50}$$

where μ is the learning rate, $w^{(n)}$ is a parameter in an arbitrary layer at iteration n, and $\frac{\partial \mathcal{J}}{\partial w^{(n)}}$ is the partial derivative of the cost function $\mathcal{J}$ with respect to the weight. Rumelhart *et al.* showed in [153] that $\frac{\partial \mathcal{J}}{\partial w^{(n)}}$ can be straightforwardly decomposed, using the chain rule of differentiation, into a chain of products of simple partial derivatives that, except for the activation function, only involved differentiation of linear functions. In fact, Rumelhart *et al.* showed that if the activation function was given by the sigmoid function defined as

$$\phi(x) = \frac{1}{1 + e^{-x}}, \quad x \in \mathbb{R}, \quad \phi(x) \in (0, 1), \tag{51}$$

where

$$\frac{\partial \phi(x)}{\partial x} = \phi(x)(1 - \phi(x)), \tag{52}$$

this chain of partial derivatives could easily and efficiently be evaluated for all parameters in a multi-layer FNN, consequently, allowing multi-layer FNNs to be trained successfully.

However, although the back-propagation technique enabled multi-layer FNNs, with sigmoid activation functions, to be efficiently trained, the sigmoid function, at the same time, prohibited multi-layer FNNs with more than a few layers to be trained due to a phenomena known as the *vanishing gradient problem*. The vanishing gradient problem occurs because $\frac{\partial \mathcal{J}}{\partial w}$ is decomposed into a chain of products where $\frac{\partial \phi(x)}{\partial x}$ appears once for each layer and since $\frac{\partial \phi(x)}{\partial x} \leq 0.25$, the partial derivative $\frac{\partial \mathcal{J}}{\partial w}$ progressively gets smaller and smaller as the number of layers increase, i.e., the gradient vanishes for FNNs with a large number of layers.

A way to alleviate this problem is to use an activation function whose first derivative is close to one. Such an activation function is the rectified linear unit (ReLU) [167], which is simply a half-wave rectifier given as $max(x, 0)$. The ReLU has the very simple first sub-derivatives given as $\frac{\partial}{\partial w} max(x, 0) = 0$ for $x < 0$ and $\frac{\partial}{\partial w} max(x, 0) = 1$ for $x > 0$, which effectively reduces the vanishing gradient problem. In fact, using the ReLU instead of the sigmoid activation function is one of the key differences that enable deep FNNs, with hundreds of layers and billions of parameters, to be trained successfully with back-propagation and without unsupervised pre-training [20, 157, 173, 198, 202, 203].

2.3 Recurrent Neural Networks

We now turn our attention to a DNN architecture known as a Recurrent Neural Network (RNN). In Sec. 2.2 we saw that a FNN is a universal function approximator and one can ask why another architecture is needed if a FNN can approximate any function. The answer is at least twofold. First, the universal approximation theorem of FNNs says only something about the representational capacity of FNNs with an unlimited number of parameters. It does not, however, say anything about the FNN topology, i.e. the number of layers and the number of units per layer. Secondly, optimization of FNNs, and DNNs in general, is a non-convex problem and generally no guaranties exist in terms of convergence and optimality, when the parameters are found using gradient descent and back-propagation [158, 204]. Therefore, some DNN architectures might be more efficient in terms of parameters, or superior in terms of performance, compared to FNNs trained using gradient descent and back-propagation, and indeed, such architectures exist. Two such popular architectures are the RNN, to be described in this section, and the CNN, which will be the topic of the next section.

Specifically, the basic single-layer RNN architecture is given as [205]

$$\underline{h}^{(n)} = \phi(\underline{\underline{W}}\underline{x}^{(n)} + \underline{\underline{V}}\underline{h}^{(n-1)} + \underline{b}), \tag{53}$$

where $\phi(\cdot)$ is an activation function, $\underline{\underline{W}}$, and $\underline{\underline{V}}$ are parameter matrices, $\underline{b}$ is a bias vector, and $\underline{h}^{(n)}$ is the output the RNN at time index n. From Eq. (53) it is seen that the RNN architecture operates with a time index (n), and differently from the FNN architecture (Eq. (47)) the RNN has a recurrent connection that is shared between time-steps, hence the name RNN. This time index and weight sharing property, is exactly what differentiates RNNs from FNNs and what allows RNNs to be efficient models of sequential data with strong temporal structure, such as speech. Also, similarly to FNN architectures, RNNs can be stacked into deep RNNs, hence increasing the model capacity [206], and it can even be shown that RNNs can model any dynamical system

with any required accuracy, which is known as the universal approximation theorem for RNNs [207]. Furthermore, as a consequence of the recurrent connection, for any finite n, i.e. $n = 1, 2, \ldots, N$, a FNN architecture exists that has the exact same behavior as a RNN [205]. In fact, this is exactly what is utilized when training RNNs using the back-propagation-through-time technique. First, the RNN is converted, or unrolled in time, into an N-layer FNN, and then, it is trained using the standard back-propagation technique as described in Sec. 2.2. [158].

However, similarly to the FNNs, if the network is deep, i.e. if N is large, training usually fails due to the vanishing gradient problem, which, in practice, prohibits the use of RNNs for signals with long time dependencies [208]. Also, since the recurrent weights are shared across time, the vanishing gradient problem is even more severe for RNNs, as the weights are constant for all time steps. This is especially true in the case where $\|\underline{\underline{V}}\| < 1$, i.e. the matrix norm of $\underline{\underline{V}}$ is less than one. A phenomena known as exploding gradients also exists, which can occur in the case when $\|\underline{\underline{V}}\| > 1$, although this is usually handled simply by clipping the gradients [158].

Several techniques have been proposed to alleviate the vanishing gradient problem. For example, by constraining the recurrent weight matrix to be orthogonal the vanishing gradient problem can be reduced (see e.g. [209–213]). A different approach to avoid the vanishing gradient problem is to change the RNN architecture into what is known as gated-RNNs [158, 214–219], and the most popular of these gated-RNNs is the Long Short-Term Memory (LSTM)-RNN [218, 219] given by

$$
\begin{aligned}
\underline{i}^{(n)} &= \sigma\left(\underline{\underline{W}}_i \underline{x}^{(n)} + \underline{\underline{V}}_i \underline{h}^{(n-1)} + \underline{b}_i\right), \\
\underline{f}^{(n)} &= \sigma\left(\underline{\underline{W}}_f \underline{x}^{(n)} + \underline{\underline{V}}_f \underline{h}^{(n-1)} + \underline{b}_f\right), \\
\underline{o}^{(n)} &= \sigma\left(\underline{\underline{W}}_o \underline{x}^{(n)} + \underline{\underline{V}}_o \underline{h}^{(n-1)} + \underline{b}_o\right), \\
\underline{d}^{(n)} &= \tanh\left(\underline{\underline{W}}_c \underline{x}^{(n)} + \underline{\underline{V}}_c \underline{h}^{(n-1)} + \underline{b}_c\right), \\
\underline{c}^{(n)} &= \underline{f}^{(n)} \circ \underline{c}^{(n-1)} + \underline{i}^{(n)} \circ \underline{d}^{(n)}, \\
\underline{h}^{(n)} &= \underline{o}^{(n)} \circ \tanh\left(\underline{c}^{(n)}\right),
\end{aligned}
\tag{54}
$$

where $\underline{x}^{(n)}$ is the input, $\sigma(\cdot)$ and $\tanh(\cdot)$ denote the sigmoid and hyperbolic tangent functions, "$\circ$" denotes element-wise multiplication, and the subscripts "$\cdot_i$", "$\cdot_f$", "$\cdot_o$", and "$\cdot_c$", with an abuse of notation, denote the parameters associated with the input gate, forget gate, output gate, and the cell state, respectively [218, 219]. The LSTM architecture minimizes the vanishing gradient problem primarily due to the algorithm step $\underline{c}^{(n)} = \underline{f}^{(n)} \circ \underline{c}^{(n-1)} + \underline{i}^{(n)} \circ \underline{d}^{(n)}$, which is known as the cell state. The cell state has a recurrent connection

to itself with no activation function. Furthermore, the value of the cell state is controlled by "gates" that only act multiplicative on the cell state, hence, controlling the flow of information into the cell and out of the cell. Since the cell state has no activation function and no weight directly associated with it, its value will remain constant during gradient updates hence, avoiding the vanishing gradient problem. However, although the LSTM given by Eq. (54), is more computational complex compared to the basic RNN given by Eq. (53), it is far easier to train, and works better in practice when N is large [220, 221]. Consequently, the LSTM-RNN architecture is currently the most popular RNN architecture used for speech processing applications such as speech recognition, enhancement, and separation [120, 169, 170, 222–227].

2.4 Convolutional Neural Networks

Similarly to the RNN, the Convolutional Neural Network (CNN) is an architecture that utilizes weight-sharing, but compared to the RNN, in a fundamentally different way [228–230]. As seen from Eqs. (53) and (54), for RNN architectures the same weight matrix is shared for each time step. This configuration is known as *tied weights* as the connections between weights and inputs are constant. Although this configuration leads to powerful models due to the universal approximation theorem, it also leads to computational demanding networks due to the dense matrix vector multiplications. If, however, a large number of weights are redundant, i.e. taking similar values, due to a certain general structure in the data, it is more efficient to reuse these parameters instead of having them stored in multiple different locations in a large matrix. This is exactly the principle behind the CNN architecture. CNNs use *untied weights* that are shared for multiple inputs. Mathematically, this corresponds to the convolution between the input signal and the parameters, hence the name CNN.

For example, in the two-dimensional case, the convolution[7] between a signal matrix $\underline{\underline{X}} \in \mathbb{R}^{J \times I}$ and a parameter matrix $\underline{\underline{W}} \in \mathbb{R}^{M \times N}$, where $M < J$ and $N < I$, is simply given as

$$s_{j,i} = \sum_{m=1}^{M} \sum_{n=1}^{N} x_{j+m,i+n} w_{m,n}, \tag{55}$$

where $x_{j,i}$ and $w_{m,n}$ denote entries j, i, and m, n of $\underline{\underline{X}}$ and $\underline{\underline{W}}$, respectively.

Similarly to the FNN and RNN architectures, a CNN also consists of multiple layers of non-linear mappings, where each layer is based on a non-linear activation function. Differently from the FNNs and RNNs, for CNNs the input to the activation function is the convolution between the input to the layer

[7] Although this is technically speaking the cross-correlation, since we use positive increments and not negative, we follow the convention and refer to it as convolution [158].

and the set of layer-specific parameters $\underline{\underline{W}}$, usually called filters or kernels. This is fundamentally different from the dense matrix vector product used by the FNN and RNN architectures.

Usually each layer consists of multiple kernels that are all convolved with the same input, hence producing a larger number of outputs than inputs, known as feature maps. To reduce the memory complexity, usually a stride larger than one is used, which means that j, i are incremented with step sizes larger than one. Another step usually applied is known as pooling, where each feature map is down-sampled with a certain factor. This pooling step adds translational invariance to the CNN, which usually is a desirable quality, while at the same time reducing the memory requirement. If the CNN is used for classification, a FNN is usually used as the output layer.

Since the number of parameters is defined by M and N and not by J and I, as with FNNs and RNNs, and since $M \ll J$ and $N \ll I$ for many practical systems, CNNs can potentially require far less parameters for the same performance, i.e. CNNs can be more parameter-efficient compared to FNNs and RNNs [158]. This is especially true for applications involving natural images where features such as edges usually contain more information about the content of the image than solid-color regions do. In such cases the kernels $\underline{\underline{W}}$ can easily extract such information with a low number of parameters using e.g. a 3×3 edge detector (e.g Sobel kernel) [231]. A FNN on the other hand, would potentially require several orders of magnitude more parameters to apply the same operation as the operation should be a matrix vector product. In fact, this is what makes CNNs a very powerful model for natural images.

It has even been shown that CNNs trained on large datasets, containing natural images, learn very specific and intuitive kernels at each layer [188, 232]. At the first layer the kernels resemble simple edge detectors. At the next layer combinations of edge-detectors are combined into more abstract, although distinct, objects and at even higher layers these abstract objects become identifiable as the different target classes in the dataset, such as animals, persons, etc [188, 232].

Finally, since the convolution in Eq. (55) is differentiable, CNNs can similarly to the FNN and RNN architectures be trained efficiently using the back-propagation technique, and today, CNNs are by far the most successful DNN architecture for image applications (see e.g. [233–235]).

3 Deep Learning for Enhancement and Separation

So far, classical non-deep learning-based methods for single-microphone speech enhancement have not been able to improve speech intelligibility of noisy speech in realistic scenarios (see Sec. 1). Similarly, classical non-deep learning-based methods for single-microphone multi-talker speech separa-

tion have so far not been able to successfully separate an audio signal consisting of multiple speech signals into individual speech signals without prior knowledge about the speakers. With the emergence of deep learning some of the challenges faced by previous techniques can now be overcome. In this section, we will review some of these deep learning-based techniques for single-microphone speech enhancement and single-microphone multi-talker speech separation.

3.1 Deep Learning Based Speech Enhancement

After Hinton *et al.* [159, 160] showed that DNNs could be trained successfully on an image-recognition task, a renewed interest in deep learning-based single-microphone speech enhancement emerged. In general, these techniques can be divided into two types: mask approximation-based techniques and signal approximation-based techniques.

3.1.1 Mask Approximation

Let $\underline{x}_m$ and $\underline{y}_m$ denote time-frame m of a time-domain clean-speech signal and noisy-speech signal, respectively. Furthermore, let $\underline{a}_m$ and $\underline{r}_m$ denote the STFT spectral magnitude vectors of $\underline{x}_m$ and $\underline{y}_m$, respectively. Also, let $h(\underline{y}_m)$ denote a feature transformation of $\underline{y}_m$. Finally, let

$$\hat{\underline{g}}_m = f_{DNN}(h(\underline{y}_m), \underline{\theta}), \tag{56}$$

denote a gain vector, estimated by a DNN[8] $f_{DNN}(\cdot,\cdot)$ with parameters $\underline{\theta}$, such that $\hat{\underline{a}}_m = \hat{\underline{g}}_m \circ \underline{r}_m$ is an estimate of the clean speech spectral magnitude $\underline{a}_m$, and $\circ$ is element-wise multiplication. The enhanced time-domain speech signal $\hat{\underline{x}}_m$ is then acquired by IDFT using the phase of the noisy signal.

The goal of the mask approximation-based technique is then to find a set of DNN parameters $\underline{\theta}^*$ such that

$$\theta^* = \underset{\underline{\theta}}{\operatorname{argmin}} \sum_{\mathcal{D}_{train}} \mathcal{J}(\hat{\underline{g}}_m, \underline{g}_m), \quad (\underline{y}_m, \underline{g}_m) \in \mathcal{D}_{train}, \tag{57}$$

where $\mathcal{D}_{train}$ denotes a training dataset, $\mathcal{J}(\cdot,\cdot)$ denotes a cost function, $\underline{g}_m$ is a target gain vector, and the dependence on $\underline{y}_m$ and $\underline{\theta}$ is implicit via $\hat{\underline{g}}_m$. That is, the mask approximation-based technique aims to minimize the difference as measured by $\mathcal{J}(\cdot,\cdot)$ between the target gain $\underline{g}_m$ and the estimated gain $\hat{\underline{g}}_m$ (see e.g. [83, 236–242]). In the following, we review the work related to two of the most popular target gains: the Ideal Binary Mask (IBM) [83] and the Ideal Ratio Mask (IRM) [236].

[8] Note, $\hat{\underline{g}}_m$ can also be a function of multiple input vectors, i.e. for multiple m, which usually leads to improved performance for feed-forward DNNs.

Ideal Binary Mask
In the STFT domain the IBM is defined as (see e.g. [243])

$$\hat{g}^{IBM}(k,m) = \begin{cases} 1 & \text{if } \frac{|x(k,m)|}{|v(k,m)|} > T_{SNR}(k) \\ 0 & \text{otherwise,} \end{cases} \tag{58}$$

where $|x(k,m)|$ and $|v(k,m)|$ denote STFT spectral magnitudes for frequency bin k and time-frame m of the clean speech signal and the noise signal, respectively, and $T_{SNR}(k)$ denote a frequency-dependent tuning parameter (see Eqs. (38), and (39)).

One of the first to use the IBM target for DNN based single-microphone speech enhancement was Wang *et al.* [244]. Wang *et al.* proposed to use FNNs to estimate the IBM from a noisy speech signal. The FNNs were first trained using the unsupervised pre-training technique and then fine-tuned using back-propagation. However, instead of using the IBM for speech enhancement, the output vector of the penultimate FNN layer was used as a feature vector for training SVMs. Using these FNN-generated feature vectors the SVMs were trained to estimate IBMs used for speech enhancement. This approach was very similar to the previous techniques from Sec. 1, where GMMs [78] and SVMs [79] were used, but since FNNs were used as feature extractors, performance improved compared to previous techniques where hand-engineered features were used [78, 79].

Motivated by Wang *et al.* [244], Healy *et al.* [245] proposed to use a FNN-estimated IBM for speech enhancement directly, i.e without using SVMs. This approach led to further improvements compared to previous systems [78, 79, 244], presumably due to an increased amount of training data. Healy *et al.* [245] even reported large improvements in speech intelligibility for both normal hearing and hearing impaired listeners in a listening test. Similar conclusions were later reported in a subsequent study with a computationally more efficient system that did not use unsupervised pre-training [246]. However, similarly to previous machine learning-based techniques [78, 79], Wang *et al.* [244] and Healy *et al.* [245] used prior knowledge generally not available in a real-life situation, as the same noise sequence was used during training and test. That is, although these systems achieved impressive performance in unrealistic conditions, they did not reveal any information about the performance to be expected in general real-life scenarios.

Ideal Ratio Mask
Although IBM-based speech enhancement systems can achieve good performance (see e.g. [78, 79]) several studies suggested (see e.g. [65, 66, 236]) that a continuous mask might perform better as the binary T-F segmentation of the IBM, as either speech or noise dominated, might be too coarse as speech and noise is likely to be present at the same time in the same T-F unit.

Obviously, since the IBM is a special case of a general continuous mask, it is expected that a continuous mask can outperform a binary mask in terms of speech enhancement evaluation metrics [65]. One such continuous mask, which highly resembles the frequency domain Wiener filter (see Eq. (12)), is the IRM defined as

$$g^{IRM}(k,m) = \frac{|x(k,m)|^{\beta}}{|x(k,m)|^{\beta} + |v(k,m)|^{\beta}}, \tag{59}$$

where $|x(k,m)|$ and $|v(k,m)|$ denote the clean-speech signal magnitude and noise signal magnitude in frequency bin k and time frame m, respectively, and β is a tuning parameter [243]. Note, unlike the Wiener filter which is statistically optimal, the IRM is not optimal in any obvious way and was presumably motivated heuristically.

In [247] the IRM was proposed as a DNN training target and it was reported that the IRM outperformed the IBM, when used in a speech enhancement front-end for an ASR system. It was later shown that the IRM also outperformed the IBM in terms of objective evaluation metrics such as PESQ and STOI, when tested in various acoustic environments [243].

Furthermore, using a similar technique, large improvements in speech intelligibility was reported in [248] for hearing impaired listeners and moderate improvements for normal hearing listeners. In fact, although the system in [248] was "narrow" in the sense that it was speaker and noise-type specific, i.e. trained and tested in matched speaker and noise type conditions, it was the first study to report significant improvements in speech intelligibility for hearing impaired and normal hearing listeners using a single-microphone speech enhancement algorithm. In a subsequent study [249], generalizability with respect to unknown noise sources was investigated and it was reported in [250] that improvements in speech intelligibility could be achieved for noise types not seen during training, if a very large number of noise types were included in the training set. However, the improvement in speech intelligibility in [250] was significantly reduced compared to [248], especially for normal hearing listeners where modest improvements were achieved for a babble noise type and practically no improvement for a cafeteria noise type. In addition to the promising results in [248, 250–252], where the intelligibility improvements were reported for normal hearing and/or hearing impaired listeners using hearing-aids, promising results have also been reported for users of cochlear implants (see e.g. [253–255]).

Even though the studies in e.g. [248, 250–255] showed promising results, they generally only considered either a single noise type, a single speaker or a narrow range of SNRs. That is, these studies only revealed information about the performance to be expected by DNN based speech enhancement systems in non-general usage scenarios where either the noise type, speaker identity or SNR is known *a priori*.

3.1.2 Signal Approximation

Differently from the mask approximation-based technique where the goal is to minimize the difference between an estimated gain and a target gain (see Eq. (57)), the goal of the signal approximation-based techniques is to minimize the difference between the clean speech, e.g. clean speech STFT magnitudes, and the estimated speech (see e.g. [243, 256–261]).

For example, in [259] it is proposed to use a cost function defined as

$$\theta^* = \underset{\underline{\theta}}{\operatorname{argmin}} \sum_{\mathcal{D}_{train}} \mathcal{J}(\underline{\hat{a}}_m, \underline{a}_m), \quad (\underline{r}_m, \underline{a}_m) \in \mathcal{D}_{train}, \tag{60}$$

where the goal is to find a gain vector $\underline{\hat{g}}_m$, which, when applied to the noisy magnitude $\underline{r}_m$ minimize the difference between the estimated speech signal magnitude $\underline{\hat{a}}_m = \underline{\hat{g}}_m \circ \underline{r}_m$ and the target signal magnitude $\underline{a}_m$. This is arguably more sensible than the mask approximation-based technique, as no target gain is explicitly defined, and the DNN is trained to estimate a gain that achieve the minimum cost with respect to the target $\underline{a}_m$, i.e. the clean speech magnitude. In fact, when training a DNN using Eq. (60) the gain that is indirectly estimated is given as

$$g^{AM}(k,m) = \frac{|x(k,m)|}{|y(k,m)|}, \tag{61}$$

where $|y(k,m)|$ denotes the noisy speech signal magnitude in frequency bin k and time frame m, which ultimately allows for perfect reconstruction of the clean speech magnitude, i.e. $\underline{a}_m = \underline{g}_m^{AM} \circ \underline{r}_m$ [259].

However, since the phase of the noisy signal is typically used for reconstructing the enhanced speech signal in the time domain, perfect reconstruction of $\underline{a}_m$ only leads to perfect time domain signal reconstruction in the case when $|y(k,m)| = |x(k,m)| + |v(k,m)|$. This, unfortunately, is only true in the unlikely event when the clean speech and noise have identical phases, i.e. $\angle x(k,m) = \angle v(k,m)$. Since $|y(k,m)| \neq |x(k,m)| + |v(k,m)|$ in general, it was proposed in [260, 261] to use the Phase Sensitive Approximation (PSA) cost function defined as

$$\theta^* = \underset{\underline{\theta}}{\operatorname{argmin}} \sum_{\mathcal{D}_{train}} \mathcal{J}(\underline{\hat{a}}_m \, , \, \underline{a}_m \circ \underline{\phi}_m), \quad (\underline{r}_m, \underline{a}_m) \in \mathcal{D}_{train}, \tag{62}$$

where $\underline{\phi}_m = [\phi(1,m), \phi(2,m), \ldots, \phi(K,m)]^T$ and $\phi(k,m) = \cos(\angle x(k,m) - \angle y(k,m))$. In fact, the PSA gain that minimizes Eq. (62) is known as the Phase Sensitive Filter (PSF) and is given by

$$g^{PSF}(k,m) = Re\left[\frac{x(k,m)}{y(k,m)}\right] = \frac{|x(k,m)|}{|y(k,m)|}\cos(\angle x(k,m) - \angle y(k,m)), \tag{63}$$

and is the optimal real-valued filter that minimizes $|g(k,m)y(k,m) - x(k,m)|$. Furthermore, except in the unlikely case when $\angle x(k,m) = \angle v(k,m)$, Eq. (63), will lead to a higher SNR compared to e.g. the IRM [260]. Finally, the PSA cost function (Eq. (62)) is currently the training objective for DNN based speech enhancement that achieves the best performance in terms of speech enhancement evaluation metrics such as PESQ and STOI [262]. However, similarly to the studies evaluating the mask approximation technique for DNN based speech enhancement, the studies based on signal approximation (e.g. [243, 256–261]) were in general trained and tested in narrow usage scenarios where either the noise type, speaker identity or SNR was known *a priori*. That is, these studies also reveal limited information about how DNN based speech enhancement systems perform in general usage scenarios.

3.2 Deep Learning Based Speech Separation

Similarly to the renewed interest in single-microphone speech enhancement (Sec. 3.1), a renewed interest for DNN based single-microphone multi-talker speech separation emerged as well. Some of the first to apply modern DNNs to single-microphone multi-talker speech separation were Du *et al.* [263] and Huang *et al.* [264, 265].

In [263] a multi-layer FNN was trained, using unsupervised pre-training, to estimate the log-power spectrum for a target speaker from the log-power spectrum of a mixed signal consisting of two speakers. This approach is very similar to the enhancement techniques in [256–258] and the main difference is the interference signal, which is a speech signal in [263] and not environmental noise signals as in [256–258]. Nevertheless, Du *et al.* [263] showed good separation performance of a known speaker in terms of STOI and output-SNR using a signal approximation-based approach.

Differently from Du *et al.* [263], Huang *et al.* [264, 265] proposed to use multi-layer RNNs to separate a two-speaker mixture signal into the original two speech signals, i.e. Huang *et al.* [264, 265] proposed to separate the two speech signals in the mixture signal, whereas Du *et al.* [263] simply extracted a known "target" speaker. Huang *et al.* [264, 265] separated a mixture signal using a multi-layer RNN with two output streams, i.e. the output vector was twice the original size. For each output stream the RNN was trained using a signal approximation-based approach to minimize the MSE between the true source signals and the separated signals. Although the system was speaker-dependent, i.e. the same speakers were used for training and testing, Huang *et al.* [264, 265] showed that this approach works well for mixture signals containing both same-gender and opposite-gender speech signals.

However, even though Du *et al.* [263] and Huang *et al.* [264, 265] showed that DNNs could be used for single-microphone multi-talker speech separation and that they outperformed other existing methods based on GMMs and

NMF, their methods were still rather limited as they, similarly to the factorial HMM-based techniques presented in Sec. 1.2.4, were speaker-dependent as they required detailed *a priori* knowledge about the speakers.

A few techniques [266, 267] did manage to overcome this speaker dependence by introducing additional assumptions. For example, in [266] it was assumed that one speech signal always had an average energy level larger than the other speech signal, i.e. a mixture SNR different from 0 dB. With this assumption, one multi-layer FNN was trained to extract the speaker with the high average energy and another multi-layer FNN was trained to extract the speech signal with low average energy. With this approach a mixture signal containing two speakers of unknown identity could be separated somewhat successfully. In [267] it was instead assumed that the mixture signal consisted of exactly one male and one female speaker. In this case, a multi-layer FNN with two output streams was trained to separate the input mixture such that the female speech signal was assigned to e.g. output stream one and the male speech signal to output stream two. In turn, this enabled the FNN to separate unknown speakers of different gender. However, even though the techniques proposed in [266, 267] could separate two-speaker speech signals without *a priori* knowledge about the identity of the speakers, they were still rather limited as they were not easily scaled to more than two speakers of various gender.

3.2.1 Label Permutation Problem

The limited success of DNN based techniques for speaker-independent multi-talker speech separation, and the reason why most techniques considered only known-two-speaker separation (see e.g. [263–273]), is partly due to a label permutation problem. When training a DNN for speaker-independent multi-talker speech separation the permutation of the sources at the output of the DNN is unknown. Specifically, for a two-speaker separation task, let $\underline{o}_1$ denote a target vector for speaker one, and let $\underline{o}_2$ denote a target vector for speaker two. Furthermore, let $\underline{o} = \left[\underline{o}_1^T \; \underline{o}_2^T\right]^T$ denote a concatenated supervector. Finally, let $\underline{\hat{o}}$, which is the output of a DNN, denote the estimate of $\underline{o}$. Then, during training, the target vector can in principle be in one of two configurations: either as $\underline{o} = \left[\underline{o}_1^T \; \underline{o}_2^T\right]^T$ or as $\underline{o} = \left[\underline{o}_2^T \; \underline{o}_1^T\right]^T$. Empirically, it has been observed that if $\underline{o}_1$ and $\underline{o}_2$ always represent the same speakers, or by speakers of the same gender, training with a predefined permutation, such as $\underline{o} = \left[\underline{o}_1^T \; \underline{o}_2^T\right]^T$ or as $\underline{o} = \left[\underline{o}_2^T \; \underline{o}_1^T\right]^T$, is possible, and is basically the technique used in [264–267, 272]. On the other hand, due to the label permutation problem, training simply fails, if the training set consist of many utterances spoken by many speakers of both genders.

3.2.2 Deep Clustering

The first successful technique, known as deep clustering, that solved the label permutation problem was proposed by Hershey *et al.* [274]. In deep clustering, the speech separation problem is cast as a clustering problem instead of a classification or regression problem as previous techniques. Hershey *et al.* [274] used LSTM-RNNs to learn a mapping from each T-F unit in the mixture signal to a high dimensional embedding space, where embeddings of T-F units belonging to the same speaker are close (in some sense) and form speaker-specific clusters, which can then be used to separate the speech signals.

More specifically, let $\underline{y} \in \mathbb{R}^N$ denote a feature vector of a mixture signal defined according to Eq. (40) containing $s = 1, 2, \ldots, S$ linearly mixed speakers. The feature representation can e.g. be given by a STFT such that $N = K \times M$ denote the total number of T-F units, where K is the total number of frequency bins and M is the total number of time-frames. Furthermore, let $\underline{\underline{V}} \in \mathbb{R}^{N \times S}$ denote a target matrix with a row for each index in $\underline{y}$, i.e. for each T-F unit, and each row in $\underline{\underline{V}}$ is given by an S-dimensional one-hot encoded vector that indicates what speaker a given T-F unit belongs to. For example, for $S = 3$, if the first T-F unit in $\underline{y}$ is dominated by, say, speaker one, the first row in $\underline{\underline{V}}$ will be given as $[1\ 0\ 0]$. The second row will be $[0\ 0\ 1]$, if the second entry in $\underline{y}$ is dominated by speaker three, and so on. The rows in $\underline{\underline{V}}$ can be viewed as a generalization of the IBM to multiple speakers as the assignment of a T-F unit to a speaker is simply defined as the speaker with the most energy in the given T-F unit. The matrix $\underline{\underline{V}}$ can also be seen as an S-dimensional embedding of each entry in $\underline{y}$ and from $\underline{\underline{V}}$ it is trivial to identify which T-F units in $\underline{y}$ belong to the same speaker, by simply applying a clustering algorithm, e.g. K-means [275], to the rows of $\underline{\underline{V}}$.

Since $\underline{\underline{V}}$ is easily constructed in laboratory conditions, where speech mixtures can be synthetically mixed according to Eq. (40), one can imagine to use $\underline{\underline{V}}$ as a training target for supervised learning and then estimate a matrix $\hat{\underline{\underline{V}}} \in \mathbb{R}^{N \times D}$ as

$$\hat{\underline{\underline{V}}} = f(\underline{y}, \theta), \tag{64}$$

where $f(\underline{y}, \theta)$ denote a parameterized learning model that maps each entry of $\underline{y}$ into a D-dimensional embedding space such that they are clustered similarly to the S-dimensional embeddings in $\underline{\underline{V}}$.

In fact, Hershey *et al.* showed in [274] that an estimate $\hat{\underline{\underline{V}}}$ can easily be acquired by a model $f(\underline{y}, \theta)$ when it has been trained using a cost function given as

$$\mathcal{J}(\hat{\underline{\underline{V}}}, \underline{\underline{V}}) = \left\| \hat{\underline{\underline{V}}} \hat{\underline{\underline{V}}}^T - \underline{\underline{V}} \underline{\underline{V}}^T \right\|_F^2, \tag{65}$$

where $\|\cdot\|_F^2$ is the squared Frobenius, and $\hat{\underline{\underline{V}}} \hat{\underline{\underline{V}}}^T \in \mathbb{R}^{N \times N}$ and $\underline{\underline{V}} \underline{\underline{V}}^T \in \mathbb{R}^{N \times N}$ denote affinity matrices that indicate if a pair of T-F units belong to the same

speaker/cluster. If an estimate $\hat{\underline{\underline{V}}}$ is acquired from a well-trained model $f(\underline{y}, \theta)$, the cluster assignments for all T-F units are easily found using e.g. K-means clustering, which can then be used to form a binary mask that can separate the mixture signal.

Note, the matrices $\hat{\underline{\underline{V}}}\hat{\underline{\underline{V}}}^T$ and $\underline{\underline{V}}\,\underline{\underline{V}}^T$ are $N \times N$, which for long signals gets intractable to compute. For example, for a 10s audio signal with a 256-point STFT using 10 ms frame hop, these matrices have more than 16 billion entries. However, as $S, D \ll N$, Hershey *et al.* proposed to minimize the equivalent, but computationally tractable, cost function given by

$$\mathcal{J}(\hat{\underline{\underline{V}}}, \underline{\underline{V}}) = \left\| \hat{\underline{\underline{V}}}^T \hat{\underline{\underline{V}}} \right\|_F^2 - 2 \left\| \hat{\underline{\underline{V}}}^T \underline{\underline{V}} \right\|_F^2 + \left\| \underline{\underline{V}}\,\underline{\underline{V}}^T \right\|_F^2, \tag{66}$$

which scales according to $\mathcal{O}(D^2)$, and not as $\mathcal{O}(N^2)$.

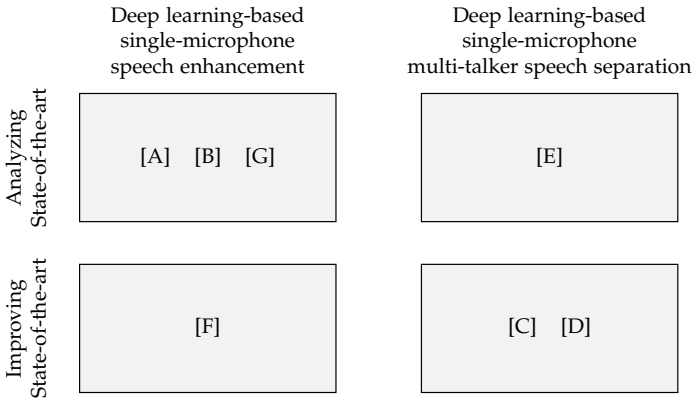
As shown in Hershey *et al.* [274], the label permutation problem is elegantly avoided when the speech separation problem is cast as a clustering problem. Furthermore, when $f(\underline{y}, \theta)$ is modeled using LSTM-RNNs, state-of-the-art results can be achieved.

However, although Hershey *et al.* [274] reported unprecedented results on a speaker-independent single-microphone multi-talker speech separation task, the deep clustering approach had several drawbacks. For example, during inference, a clustering algorithm, e.g. K-means [275], is required to separate the speakers and consequently the number of speakers S needs to be known *a priori*. Also, deep clustering as proposed by Hershey *et al.* [274] use a binary gain, which may not be optimal if a large number of speakers or noise sources are present in the mixture. Furthermore, in [274] only clean speech is considered and it is not obvious how noise sources should be handled. Finally, as each T-F unit is represented by a D-dimensional embedding vector (in [274] $D \approx 40$), the output of a deep clustering model needs to be D-times larger than the input, which might be computationally demanding for long signals.

As a final note, concurrently with the work presented in this thesis, the deep clustering technique has been improved in several aspects such as, soft-clustering [276], regression based speech enhancement [277], improved objective functions [278], and phase estimation [279], which have led to significant gains in performance when measured by SDR (see Sec. 1.3.3). Also, concurrently with our work, other competing techniques have been proposed such as the deep attractor network [280, 281] and source-contrastive estimation [282], which are both techniques inspired by deep clustering.

4 Scientific Contribution

The main body of this thesis (Part II) consists of a collection of seven papers. These papers have contributed scientifically by analyzing state-of-the-art techniques, leading to novel insights, or improving state-of-the-art techniques, with novel algorithms, within two disciplines: Deep learning-based single-microphone speech enhancement and deep learning-based single-microphone multi-talker speech separation. Figure 3 summarizes for each of the seven papers the type of scientific contribution and the discipline within which the contribution is made.



[A] "Speech Intelligibility Potential of General and Specialized Deep Neural Network Based Speech Enhancement Systems", *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, January 2017.

[B] "Speech Enhancement Using Long Short-Term Memory Based Recurrent Neural Networks for Noise Robust Speaker Verification", *IEEE Spoken Language Technology Workshop*, December 2016.

[C] "Permutation Invariant Training of Deep Models for Speaker-Independent Multi-talker Speech Separation", *IEEE International Conference on Acoustics, Speech, and Signal Processing*, March 2017.

[D] "Multi-talker Speech Separation With Utterance-Level Permutation Invariant Training of Deep Recurrent Neural Networks", *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, October 2017.

[E] "Joint Separation and Denoising of Noisy Multi-talker Speech Using Recurrent Neural Networks and Permutation Invariant Training", *IEEE International Workshop on Machine Learning for Signal Processing*, September 2017.

[F] "Monaural Speech Enhancement Using Deep Neural Networks by Maximizing a Short-Time Objective Intelligibility Measure", *IEEE International Conference on Acoustics, Speech, and Signal Processing*, April 2018.

[G] "On the Relationship between Short-Time Objective Intelligibility and Short-Time Spectral-Amplitude Mean Squared Error for Speech Enhancement", *under major revision in IEEE/ACM Transactions on Audio, Speech, and Language Processing*, August 2018.

Fig. 3: Scientific contribution of the papers making up this thesis: 1) Papers [A], [B], and [G] analyze state-of-the-art speech enhancement and contribute with novel insights. 2) Paper [F] improves state-of-the-art speech enhancement with a novel algorithm. 3) Paper [E] analyzes state-of-the-art multi-talker speech separation and contributes with novel insights. 4) Papers [C] and [D] improve state-of-the-art multi-talker speech separation with novel algorithms.

4.1 Specific Contributions

In the following, we shortly summarize the main scientific contribution of each paper in Part II.

[A] Speech Intelligibility Potential of General and Specialized Deep Neural Network Based Speech Enhancement Systems

In this paper, we study the generalizability capability of deep learning-based single-microphone speech enhancement algorithms. Specifically, we investigate how speech enhancement systems based on FNNs perform in terms of PESQ and STOI when tested in acoustic scenarios that are either matched or unmatched, i.e. the noise type, speaker, or SNR used for testing are either similar or different from the noise type, speaker or SNR used for training. This is motivated by recent studies where large improvement in PESQ and STOI have been reported by DNN based speech enhancement systems that are narrowly trained.

Not surprisingly, we find that one generally loses performance when a system that is trained in a narrow acoustic setting is tested in a more general and realistic acoustic scenario. We also find that matching the noise type is the most critical for acquiring good speech enhancement performance, whereas matching the SNR is less critical and good performance for unmatched speakers can be achieved if only a modest number of speakers are included in the training set.

[B] Speech Enhancement Using Long Short-Term Memory Based Recurrent Neural Networks for Noise Robust Speaker Verification

In this paper, we study the generalizability capability of a deep learning-based speech enhancement algorithm with respect to noise robust speaker verification. Specifically, we propose to use a LSTM-RNN based speech enhancement algorithm as a denoising front-end for a noise robust and male-speaker-independent speaker verification system.

Compared to two baseline systems based on a STSA-MMSE estimator and NMF, we find that the denoising front-end based on the LSTM-RNN performed the best in terms of equal error rate on a speaker verification task, when tested using various noise types and SNRs. Despite the fact that the LSTM-RNN was tested in unmatched male-speaker and noise type conditions, it outperformed the NMF based baseline even though this baseline utilized *a priori* information about both the speaker and noise type.

[C] Permutation Invariant Training of Deep Models for Speaker-Independent Multi-talker Speech Separation

In this paper, we propose a deep learning-based technique for single-microphone speaker-independent multi-talker speech separation. Specifically, we propose the Permutation Invariant Training (PIT) technique, which circumvent the label permutation problem, mentioned in Sec. 3.2, and allow DNNs to be trained successfully for speaker-independent multi-talker speech separation. We evaluate PIT using FNNs and CNNs for two-talker speech separation using both matched speakers, i.e. same speakers for training and test, and unmatched speakers, i.e. different speakers for training and test.

We find that FNNs and CNNs trained with PIT and tested on a speaker-independent two-talker speech separation task achieved state-of-the-art results, and outperformed techniques based on CASA and NMF in terms of SDR. We also find that CNNs trained with PIT perform on par with the deep clustering technique proposed in [274], although the PIT models are computationally less complex. Finally, we find that models trained with PIT generalize well to a Danish dataset, although the models have only been trained on English speech.

[D] Multi-talker Speech Separation With Utterance-Level Permutation Invariant Training of Deep Recurrent Neural Networks

In this paper, we propose the utterance-level Permutation Invariant Training (uPIT) technique, which is an extension of the PIT technique proposed in [C]. Although PIT allowed for deep learning-based speaker-independent multi-talker speech separation, PIT only works well in practice, when large signal contexts, i.e. a large number of frames, are used. This is a limitation for applications that require low latency. The uPIT technique, on the other hand, does not have this drawback.

In this paper we show that LSTM-RNNs trained using uPIT can achieve state-of-the-art results on a speaker-independent multi-talker speech separation task with both two-speaker and three-speaker mixed speech. Furthermore, we show that these models can achieve this performance using a small signal context of one frame, which is a dramatic reduction compared to the 50 or 100 frames required by PIT. Also, similarly to PIT, uPIT perform on par with deep clustering even though uPIT is algorithmically much simpler than deep clustering. Finally, we show that a single LSTM-RNN successfully separates both two-speaker and three-speaker mixed speech without *a priori* knowledge about the number of speakers.

[E] Joint Separation and Denoising of Noisy Multi-talker Speech Using Recurrent Neural Networks and Permutation Invariant Training

In this paper, we study aspects of the uPIT technique. Specifically, we investigate how uPIT can be used for single-microphone speaker-independent multi-talker speech separation and enhancement, simultaneously. This is different from previously speech enhancement techniques (see Sec. 3.1) that only consider a single target speaker and existing speech separation techniques (see Sec. 3.2), that consider only noise-free multi-talker mixtures.

We show that LSTM-RNNs trained using uPIT in noisy environments can improve SDR as well as ESTOI, on a speaker-independent multi-talker speech separation and enhancement task, for various noise types and SNRs. We also show that a LSTM-RNN trained using uPIT generalize well to unmatched noise types and that a single model is capable of handling multiple noise types with only a slight decrease in performance. Finally, we show that a LSTM-RNN trained using uPIT can improve both SDR and ESTOI without *a priori* knowledge about the exact number of speakers.

[F] Monaural Speech Enhancement Using Deep Neural Networks by Maximizing a Short-Time Objective Intelligibility Measure

In this paper, we propose to use a STOI inspired cost function for training DNNs for single-microphone speech enhancement. Since STOI has proven an accurate estimator of speech intelligibility, it is hypothesized that a DNN that is trained to estimate speech that maximizes STOI, might lead to speech with a large speech intelligibility. Specifically, compared to the standard STSA-MSE cost function, which does not have any obvious link to speech intelligibility, a cost function inspired by STOI might be advantageous.

We show that FNNs, trained with an approximate-STOI cost function, improve STOI when tested using matched and unmatched noise types, at multiple SNRs. More surprisingly, we observe that approximate-STOI optimal FNNs perform on par with FNNs based on the standard STSA-MSE cost function. Consequently, our results suggest that DNN based speech enhancement algorithms, based on the STSA-MSE cost function, might be essentially optimal in terms of estimated speech intelligibility as measured by STOI.

[G] On the Relationship between Short-Time Objective Intelligibility and Short-Time Spectral-Amplitude Mean Squared Error for Speech Enhancement

In this paper, we analyze the surprising result of paper [F], where no apparent gain in STOI can be achieved by a FNN based speech enhancement system trained to maximize an approximate-STOI cost function, as compared to a system trained to minimize the classical STSA-MSE cost function.

We show theoretically that the optimal Bayesian estimator that maximizes approximate-STOI, under certain general conditions, is asymptotically equivalent to the well-known STSA minimum mean square error estimator. Furthermore, through simulation experiments, we show that equality holds for practical FNN based speech enhancement systems. The theoretical and empirical results presented in this paper support the surprising result in paper [F] and optimizing for STSA-MSE leads to enhanced speech signals which are essentially optimal in terms of STOI.

4.2 Summary of Contributions

The main scientific outcomes of this thesis may be summarized as follows:

1) In papers [A] and [B], in-depth empirical analysis of the generalizability capability of modern deep learning-based single-microphone speech enhancement algorithms have been conducted. In paper [A] it is found, not surprisingly, that performance generally decreases as the acoustic variability of the test data increases. However, it is found that good generalizability with respect to unmatched speakers can be achieved if a modest amount of speakers are included in the training set. Furthermore, in paper [B] it is shown that a DNN based speech enhancement system can generalize to unmatched speakers and achieve state-of-the-art speaker verification performance without *a priori* knowledge about the speakers. The findings in papers [A] and [B], besides from contributing with novel insights, can serve as guidelines in speech-enhancement-algorithm selection or in the design process of future deep learning-based speech enhancement systems.

2) In papers [C], [D], and [E], state-of-the-art techniques for deep learning-based single-microphone speaker-independent multi-talker speech separation were proposed. Specifically, the uPIT technique was proposed, which is algorithmically simpler, yet perform on par with state-of-the-art techniques, such as deep clustering and the deep attractor network. Furthermore, uPIT easily extends to multiple speakers and works well for joint speech separation and enhancement without explicit *a priori* knowledge about the noise type or number of speakers, which, at the time of writing, is a capability only shown by uPIT.

3) In papers [F] and [G], it was hypothesized that DNN based speech enhancement systems, trained with an approximate-STOI cost function, would lead to estimated speech signals with an improved STOI score, compared to speech signals estimated by systems trained with the standard STSA-MSE cost function. However, supported by experimental and theoretical results, we conclude that this is not the case. In fact, STSA-MSE leads to enhanced speech signals which are essentially optimal in terms of STOI, and additional improvements in STOI cannot be achieved by a STOI inspired cost function.

5 Directions of Future Research

During the last decade deep learning has evolved from a somewhat exotic academic discipline to a fairly mature technology that is widely accessible and is used at an industrial scale. Consequently, deep learning has received a tremendous amount of attention from both academia and industry and new deep learning theory, and applications of deep learning, are constantly being proposed. This also applies within the areas of speech enhancement and speech separation where an almost countless number of papers have been published over the last couple of years. Promising research directions in the area of deep learning based speech enhancement and separation include:

Scale Up While Scaling Down

As apparent from Secs. 2 and 3, DNNs must be fairly big, have multiple layers, a large amount of units, and be trained on a large amount of data before they can perform well. Obviously, one way to achieve better performance is simply to scale up and train with even more data and use even larger models [156, 157]. Indeed, this is a valid approach, and is one of the main innovations, if you will, behind many state-of-the-art deep learning based techniques. However, training such models is computationally demanding, but more importantly, the memory and computational requirements of DNNs might prohibit their use in applications where computational resources are limited, such as in small embedded devices like mobile phones or hearing aids. Therefore, a direction of future research, which is already very active, is on scaling down DNNs without compromising performance, e.g. by reducing the number of parameters in an informed way, increasing the number of layers, while decreasing the number of units, to make the model more parameter-efficient, or reducing the numerical precision of the weights (see e.g. [283–289]).

Beyond Single-Microphone Algorithms

In this thesis, we have focused purely on single-microphone algorithms. However, utilizing information from multiple microphones can be beneficial if the signal of interest is spatially separated from the interference signals. In such situations, improved performance might be achieved if this information is included. Hence, a direction of future research is to study how signals from multiple microphones can be efficiently utilized in a deep learning framework. For example, one might extend the uPIT technique to work with multi-microphone signals. Several promising techniques have already been proposed, where DNNs are used in combination with multi-microphone techniques such as beamforming (see e.g. [290–300]).

Beyond Single-Modality Algorithms

It is well-known that human auditory perception is strongly influenced by visual perception [301] and that speech intelligibility in noisy acoustic conditions increase if the listener can observe the face of the person who speaks [302]. Indeed, speech enhancement and separation algorithms can also benefit from such information (see e.g. [115]), and although fusing signals from multiple modalities is a challenging task, the emergence of deep learning has alleviated some of these challenges (see e.g. [303–305]). Therefore, studying how deep learning based techniques for speech enhancement and separation can benefit from e.g. visual data is an interesting direction for future research.

Beyond the Mean Squared Error Cost Function

We have already shown in papers [F] and [G] (see Fig. 3) that an approximate-STOI cost function is equivalent to the STSA-MSE cost function and no gain in terms of STOI can be achieved by maximizing an approximate-STOI cost function. This conclusion is supported by other very recent work [306–309]. However, it might be that improvements in speech intelligibility or quality can be achieved by optimizing other perceptually-inspired cost functions such as PESQ or Binaural STOI [310], e.g. using deep reinforcement learning techniques (see e.g. [311, 312]). Therefore, an interesting direction of future research is to consider alternatives to the commonly used STSA-MSE cost function, which might lead to improved performance of DNN based speech enhancement systems.

Towards Time-Domain End-to-End Systems

One of the main advantages of deep learning-based techniques is that they do not require highly specialized, and hand-engineered, features as previous machine learning-based technique did. Today, the most used feature, and target, for speech processing applications is the STSA or log-STSA and most speech enhancement and separation algorithms apply the phase of the noisy signal for time-domain reconstruction. Obviously, this is sub-optimal since the noisy phase can only lead to distortions. Therefore, an interesting direction of future research is to study how deep learning models can operate directly on the time-domain signal[9], which potentially can lead to improved performance over the current methods which rely on the noisy phase. In fact, a potential deep learning model for time-domain processing is the CNN, which has already shown promising results with respect to time-domain speech enhancement (see e.g. [271, 290, 308, 313–317]).

[9] In fact, this is exactly what Tamura *et al.* [74] attempted in 1988 using DNNs, although, today, we might have the model architecture, data, and computational resources to actually succeed.

References

[1] H. Helmholtz, *On the Sensations of Tone*. Dover, 1954.

[2] G. Fant, *Acoustic Theory of Speech Production: With Calculations based on X-Ray Studies of Russian Articulations*. De Gruyter Inc., 1960.

[3] J. L. Flanagan, *Speech Analysis Synthesis and Perception*. Springer, 1972.

[4] J. Schnupp, E. Nelken, and A. King, *Auditory Neuroscience - Making Sense of Sound*. MIT Press, 2011.

[5] B. Moore, *An Introduction to the Psychology of Hearing*. Brill, 2013.

[6] W. T. Fitch, "The evolution of speech: a comparative review," *Trends in Cognitive Sciences*, vol. 4, no. 7, pp. 258–267, 2000.

[7] Y. Bitterman, R. Mukamel, R. Malach, I. Fried, and I. Nelken, "Ultra-fine frequency tuning revealed in single neurons of human auditory cortex," *Nature*, vol. 451, no. 7175, pp. 197–201, 2008.

[8] E. C. Cherry, "Some Experiments on the Recognition of Speech, with One and with Two Ears," *The Journal of the Acoustical Society of America*, vol. 25, no. 5, pp. 975–979, 1953.

[9] A. S. Bregman, *Auditory Scene Analysis - The Perceptual Organization of Sound*. MIT Press, 1990.

[10] A. W. Bronkhorst, "The Cocktail Party Phenomenon: A Review of Research on Speech Intelligibility in Multiple-Talker Conditions," *Acta Acustica united with Acustica*, vol. 86, no. 1, pp. 117–128, 2000.

[11] P. Divenyi, *Speech Separation by Humans and Machines*. Springer, 2005.

[12] D. Wang and G. J. Brown, *Computational Auditory Scene Analysis: Principles, Algorithms, and Applications*. Wiley-IEEE Press, 2006.

[13] J. H. McDermott, "The cocktail party problem," *Current Biology*, vol. 19, no. 22, pp. R1024–R1027, 2009.

[14] N. Mesgarani and E. F. Chang, "Selective cortical representation of attended speaker in multi-talker speech perception," *Nature*, vol. 485, no. 7397, pp. 233–236, 2012.

[15] E. Zion Golumbic *et al.*, "Mechanisms Underlying Selective Neuronal Tracking of Attended Speech at a "Cocktail Party"," *Neuron*, vol. 77, no. 5, pp. 980–991, 2013.

[16] A. W. Bronkhorst, "The cocktail-party problem revisited: early processing and selection of multi-talker speech," *Attention, Perception & Psychophysics*, vol. 77, no. 5, pp. 1465–1487, 2015.

[17] D. Wang and J. Chen, "Supervised Speech Separation Based on Deep Learning: An Overview," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 10, pp. 1702–1726, 2018.

[18] S. Haykin and Z. Chen, "The Cocktail Party Problem," *Neural Comput.*, vol. 17, no. 9, pp. 1875–1902, 2005.

[19] D. Wang, "Deep learning reinvents the hearing aid," *IEEE Spectrum*, vol. 54, no. 3, pp. 32–37, 2017.

[20] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[21] J. R. Deller, Jr., J. G. Proakis, and J. H. Hansen, *Discrete-Time Processing of Speech Signals*. Wiley-IEEE Press, 1993.

[22] P. C. Loizou, *Speech Enhancement: Theory and Practice*. CRC Press, 2013.

[23] R. C. Hendriks, T. Gerkmann, and J. Jensen, "DFT-Domain Based Single-Microphone Noise Reduction for Speech Enhancement: A Survey of the State of the Art," *Synthesis Lectures on Speech and Audio Processing*, vol. 9, no. 1, pp. 1–80, 2013.

[24] M. Brandstein and D. Ward, Eds., *Microphone Arrays: Signal Processing Techniques and Applications*, ser. Digital Signal Processing. Springer, 2001.

[25] J. Allen, "Short term spectral analysis, synthesis, and modification by discrete Fourier transform," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 25, no. 3, pp. 235–238, 1977.

[26] S. Boll, "Suppression of acoustic noise in speech using spectral subtraction," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 27, no. 2, pp. 113–120, 1979.

[27] R. McAulay and M. Malpass, "Speech enhancement using a soft-decision noise suppression filter," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 28, no. 2, pp. 137–145, 1980.

[28] Y. Ephraim and D. Malah, "Speech enhancement using a minimum-mean square error short-time spectral amplitude estimator," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 32, no. 6, pp. 1109–1121, 1984.

[29] J. H. L. Hansen and M. A. Clements, "Use of objective speech quality measures in selecting effective spectral estimation techniques for speech enhancement," in *Proc. MWSCAS*, 1989, pp. 105–108.

[30] M. M. Sondhi, C. E. Schmidt, and L. R. Rabiner, "Improving the quality of a noisy speech signal," *The Bell System Technical Journal*, vol. 60, no. 8, pp. 1847–1859, 1981.

[31] X. Haitian, Z.-H. Tan, P. Dalsgaard, and B. Lindberg, "Spectral Subtraction with Full-Wave Rectification and Likelihood Controlled Instantaneous Noise Estimation for Robust Speech Recognition," in *Proc. INTERSPEECH*, 2004, pp. 2085–2088.

[32] J. Hansen and M. Clements, "Iterative speech enhancement with spectral constraints," in *Proc. ICASSP*, vol. 12, 1987, pp. 189–192.

[33] J. Chen, J. Benesty, Y. Huang, and S. Doclo, "New Insights into the Noise Reduction Wiener Filter," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 4, pp. 1218–1234, 2006.

[34] Y. Ephraim, "Statistical-model-based speech enhancement systems," *Proceedings of the IEEE*, vol. 80, no. 10, pp. 1526–1555, 1992.

[35] J. S. Erkelens, R. C. Hendriks, R. Heusdens, and J. Jensen, "Minimum Mean-Square Error Estimation of Discrete Fourier Coefficients With Generalized Gamma Priors," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 6, pp. 1741–1752, 2007.

[36] P. C. Loizou, "Speech Enhancement Based on Perceptually Motivated Bayesian Estimators of the Magnitude Spectrum," *IEEE Transactions on Speech and Audio Processing*, vol. 13, no. 5, pp. 857–869, 2005.

[37] S. V, N. Sivaprasad, and T. K. Kumar, "A Survey on Statistical Based Single Channel Speech Enhancement Techniques," *International Journal of Intelligent Systems and Applications*, vol. 6, no. 12, p. 69, 2014.

[38] Y. Ephraim and D. Malah, "Speech enhancement using a minimum mean-square error log-spectral amplitude estimator," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 33, no. 2, pp. 443–445, 1985.

[39] N. Wiener, *Extrapolation, Interpolation, and Smoothing of Stationary Time Series*. The MIT Press, 1949.

[40] J. G. Proakis and D. G. Manolakis, *Digital Signal Processing - Principles, Algorithms, and Applications*, 4th ed. Pearson, 2007.

[41] T. Gerkmann, M. Krawczyk, and R. Rehr, "Phase estimation in speech enhancement - Unimportant, important, or impossible?" in *Proc. CEEEI*, 2012, pp. 1–5.

[42] D. Wang and J. Lim, "The unimportance of phase in speech enhancement," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 30, no. 4, pp. 679–681, 1982.

[43] K. Paliwal, K. Wójcicki, and B. Shannon, "The importance of phase in speech enhancement," *Speech Communication*, vol. 53, no. 4, pp. 465–494, 2011.

[44] T. Gerkmann, M. Krawczyk-Becker, and J. L. Roux, "Phase Processing for Single-Channel Speech Enhancement: History and recent advances," *IEEE Signal Processing Magazine*, vol. 32, no. 2, pp. 55–66, 2015.

[45] P. Mowlaee, R. Saeidi, and Y. Stylianou, "Phase importance in speech processing applications," in *Proc. INTERSPEECH*, 2014, pp. 1623–1627.

[46] P. Mowlaee, J. Kulmer, J. Stahl, and F. Mayer, *Phase-Aware Signal Processing in Speech Communication: Theory and Practice*. Wiley, 2016.

[47] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*. Prentice Hall, 2010.

[48] Y. Hu and P. C. Loizou, "Subjective Comparison of Speech Enhancement Algorithms," in *Proc. ICASSP*, 2006, pp. 153–156.

[49] R. Martin, "Speech Enhancement Based on Minimum Mean-Square Error Estimation and Supergaussian Priors," *IEEE Transactions on Speech and Audio Processing*, vol. 13, no. 5, pp. 845–856, 2005.

[50] R. C. Hendriks, R. Heusdens, and J. Jensen, "Log-Spectral Magnitude MMSE Estimators under Super-Gaussian Densities," in *Proc. INTERSPEECH*, 2009, pp. 1319–1322.

[51] J. Erkelens, R. Hendriks, and R. Heusdens, "On the Estimation of Complex Speech DFT Coefficients Without Assuming Independent Real and Imaginary Parts," *IEEE Signal Processing Letters*, vol. 15, pp. 213–216, 2008.

[52] I. Cohen, "Relaxed Statistical Model for Speech Enhancement and a Priori SNR Estimation," *IEEE Transactions on Speech and Audio Processing*, vol. 13, no. 5, pp. 870–881, 2005.

[53] ——, "Speech enhancement using super-Gaussian speech models and noncausal a priori SNR estimation," *Speech Communication*, vol. 47, no. 3, pp. 336–350, 2005.

[54] Y. Ephraim and H. L. V. Trees, "A signal subspace approach for speech enhancement," *IEEE Transactions on Speech and Audio Processing*, vol. 3, no. 4, pp. 251–266, 1995.

[55] J. Benesty, S. Makino, and J. Chen, Eds., *Speech Enhancement*, ser. Signals and Communication Technology. Springer, 2005.

[56] K. Hermus, P. Wambacq, and H. V. Hamme, "A Review of Signal Subspace Speech Enhancement and Its Application to Noise Robust Speech Recognition," *EURASIP Journal on Advances in Signal Processing*, vol. 2007, no. 1, pp. 1–15, 2006.

[57] H. Lev-Ari and Y. Ephraim, "Extension of the signal subspace speech enhancement approach to colored noise," *IEEE Signal Processing Letters*, vol. 10, no. 4, pp. 104–106, 2003.

[58] P. C. Loizou, A. Lobo, and Y. Hu, "Subspace algorithms for noise reduction in cochlear implants," *The Journal of the Acoustical Society of America*, vol. 118, no. 5, pp. 2791–2793, 2005.

[59] J. S. Lim and A. V. Oppenheim, "Enhancement and bandwidth compression of noisy speech," *Proceedings of the IEEE*, vol. 67, no. 12, pp. 1586–1604, 1979.

[60] G. Kim and P. C. Loizou, "Gain-induced speech distortions and the absence of intelligibility benefit with existing noise-reduction algorithms," *The Journal of the Acoustical Society of America*, vol. 130, no. 3, pp. 1581–1596, 2011.

[61] Y. Hu and P. C. Loizou, "A comparative intelligibility study of single-microphone noise reduction algorithms," *The Journal of the Acoustical Society of America*, vol. 122, no. 3, pp. 1777–1786, 2007.

[62] ——, "A Comparative Intelligibility Study of Speech Enhancement Algorithms," in *Proc. ICASSP*, 2007, pp. 561–564.

[63] H. Luts *et al.*, "Multicenter evaluation of signal enhancement algorithms for hearing aids," *The Journal of the Acoustical Society of America*, vol. 127, no. 3, pp. 1491–1505, 2010.

[64] P. C. Loizou and G. Kim, "Reasons why Current Speech-Enhancement Algorithms do not Improve Speech Intelligibility and Suggested Solutions," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 1, pp. 47–56, 2011.

[65] J. Jensen and R. Hendriks, "Spectral Magnitude Minimum Mean-Square Error Estimation Using Binary and Continuous Gain Functions," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 1, pp. 92–102, 2012.

[66] N. Madhu, A. Spriet, S. Jansen, R. Koning, and J. Wouters, "The Potential for Speech Intelligibility Improvement Using the Ideal Binary Mask and the Ideal Wiener Filter in Single Channel Noise Reduction Systems: Application to Auditory Prostheses," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 1, pp. 63–72, 2013.

[67] I. Brons, R. Houben, and W. A. Dreschler, "Effects of Noise Reduction on Speech Intelligibility, Perceived Listening Effort, and Personal Preference in Hearing-Impaired Listeners," *Trends in Hearing*, vol. 18, pp. 1–10, 2014.

[68] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer, 2006.

[69] G. J. McLachlan and K. E. Basford, *Mixture models : inference and applications to clustering*. Marcel Dekker, 1988, vol. 84.

[70] K. Hornik, M. Stinchcombe, and H. White, "Multilayer feedforward networks are universal approximators," *Neural Networks*, vol. 2, no. 5, pp. 359–366, 1989.

[71] G. Cybenko, "Approximation by superpositions of a sigmoidal function," *Mathematics of Control, Signals and Systems*, vol. 2, no. 4, pp. 303–314, 1989.

[72] C. Cortes and V. Vapnik, "Support-Vector Networks," *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995.

[73] B. Hammer and K. Gersmann, "A Note on the Universal Approximation Capability of Support Vector Machines," *Neural Processing Letters*, vol. 17, no. 1, pp. 43–53, 2003.

[74] S. Tamura and A. Waibel, "Noise reduction using connectionist models," in *Proc. ICASSP*, 1988, pp. 553–556.

[75] S. Tamura, "An analysis of a noise reduction neural network," in *Proc. ICASSP*, 1989, pp. 2001–2004.

[76] S. Tamura and M. Nakamura, "Improvements to the noise reduction neural network," in *Proc. ICASSP*, 1990, pp. 825–828.

[77] M. W. White, R. M. Holdaway, Y. Guo, and J. J. Paulos, "New strategies for improving speech enhancement," *International Journal of Bio-Medical Computing*, vol. 25, no. 2, pp. 101–124, 1990.

[78] G. Kim, Y. Lu, Y. Hu, and P. C. Loizou, "An algorithm that improves speech intelligibility in noise for normal-hearing listeners," *The Journal of the Acoustical Society of America*, vol. 126, no. 3, pp. 1486–1494, 2009.

[79] K. Han and D. Wang, "A classification based approach to speech segregation," *The Journal of the Acoustical Society of America*, vol. 132, no. 5, pp. 3475–3483, 2012.

[80] J. L. Roux, S. Watanabe, and J. R. Hershey, "Ensemble learning for speech enhancement," in *Proc. ASPAA*, 2013, pp. 1–4.

[81] R. D. Patterson, K. Robinson, J. Holdsworth, D. Mckeown, C. Zhang, and M. Allerhand, "Complex sounds and auditory images," in *In Proc. International Symposium on Hearing*, 1992, pp. 429–446.

[82] J. Tchorz and B. Kollmeier, "SNR estimation based on amplitude modulation analysis with applications to noise suppression," *IEEE Transactions on Speech and Audio Processing*, vol. 11, no. 3, pp. 184–192, 2003.

[83] D. Wang, "On Ideal Binary Mask As the Computational Goal of Auditory Scene Analysis," in *Speech Separation by Humans and Machines*, P. Divenyi, Ed. Springer, 2005, pp. 181–197.

[84] D. S. Brungart, P. S. Chang, B. D. Simpson, and D. Wang, "Isolating the energetic component of speech-on-speech masking with ideal time-frequency segregation," *The Journal of the Acoustical Society of America*, vol. 120, no. 6, pp. 4007–4018, 2006.

[85] U. Kjems, J. B. Boldt, M. S. Pedersen, T. Lunner, and D. Wang, "Role of mask pattern in intelligibility of ideal binary-masked noisy speech," *The Journal of the Acoustical Society of America*, vol. 126, no. 3, pp. 1415–1426, 2009.

[86] Y. Wang, K. Han, and D. Wang, "Exploring Monaural Features for Classification-Based Speech Segregation," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 2, pp. 270–279, 2013.

[87] T. May and T. Dau, "Requirements for the evaluation of computational speech segregation systems," *The Journal of the Acoustical Society of America*, vol. 136, no. 6, pp. 398–404, 2014.

[88] T. W. Parsons, "Separation of speech from interfering speech by means of harmonic selection," *The Journal of the Acoustical Society of America*, vol. 60, no. 4, pp. 911–918, 1976.

[89] B. Hanson, D. Wong, and B. Juang, "Speech enhancement with harmonic synthesis," in *Proc. ICASSP*, vol. 8, 1983, pp. 1122–1125.

[90] B. Hanson and D. Wong, "The harmonic magnitude suppression (EMS) technique for intelligibility enhancement in the presence of interfering speech," in *Proc. ICASSP*, vol. 9, 1984, pp. 65–68.

[91] T. F. Quatieri and R. G. Danisewicz, "An approach to co-channel talker interference suppression using a sinusoidal model for speech," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 38, no. 1, pp. 56–69, 1990.

[92] J. H. L. Hansen, "Speech enhancement employing adaptive boundary detection and morphological based spectral constraints," in *Proc. ICASSP*, 1991, pp. 901–904.

[93] G. J. Brown and M. Cooke, "Computational auditory scene analysis," *Computer Speech & Language*, vol. 8, no. 4, pp. 297–336, 1994.

[94] G. Hu and D. Wang, "Monaural speech segregation based on pitch tracking and amplitude modulation," *IEEE Transactions on Neural Networks*, vol. 15, no. 5, pp. 1135–1150, 2004.

[95] F. R. Bach and M. I. Jordan, "Blind One-microphone Speech Separation: A Spectral Learning Approach," in *Advances in Neural Information Processing Systems 17*, L. K. Saul, Y. Weiss, and L. Bottou, Eds. MIT Press, 2005, pp. 65–72.

[96] G. Hu and D. Wang, "Auditory Segmentation Based on Onset and Offset Analysis," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 2, pp. 396–405, 2007.

[97] ——, "A Tandem Algorithm for Pitch Estimation and Voiced Speech Segregation," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 8, pp. 2067–2079, 2010.

[98] K. Hu and D. Wang, "An Unsupervised Approach to Cochannel Speech Separation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 1, pp. 122–131, 2013.

[99] M. Cooke, J. R. Hershey, and S. J. Rennie, "Monaural Speech Separation and Recognition Challenge," *Computer Speech and Language*, vol. 24, no. 1, pp. 1–15, 2010.

[100] D. D. Lee and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, no. 6755, pp. 788–791, 1999.

[101] ——, "Algorithms for Non-negative Matrix Factorization," in *Proc. NIPS*, 2000, pp. 556–562.

[102] S. Behnke, "Discovering hierarchical speech features using convolutional non-negative matrix factorization - Semantic Scholar," in *Proc. IJCNN*, 2003, pp. 2758 – 2763.

[103] M. N. Schmidt and R. K. Olsson, "Single-Channel Speech Separation using Sparse Non-Negative Matrix Factorization," in *Proc. INTERSPEECH*, 2006, pp. 2614–2617.

[104] M. N. Schmidt and M. Mørup, "Nonnegative Matrix Factor 2-D Deconvolution for Blind Single Channel Source Separation," in *Independent Component Analysis and Blind Signal Separation*, ser. Lecture Notes in Computer Science. Springer, 2006, pp. 700–707.

[105] T. Virtanen, "Monaural Sound Source Separation by Nonnegative Matrix Factorization With Temporal Continuity and Sparseness Criteria," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 3, pp. 1066–1074, 2007.

[106] P. Smaragdis, "Convolutive Speech Bases and Their Application to Supervised Speech Separation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 1, pp. 1–12, 2007.

[107] N. Bæk Thomsen, D. Alexander Lehmann Thomsen, Z.-H. Tan, B. Lindberg, and S. Holdt Jensen, "Speaker-Dependent Dictionary-based Speech Enhancement for Text-Dependent Speaker Verification," in *Proc. INTERSPEECH*, 2016.

[108] M. N. Schmidt, J. Larsen, and F. T. Hsiao, "Wind Noise Reduction using Non-Negative Sparse Coding," in *Proc. MLSP*, 2007, pp. 431–436.

[109] J. L. Roux, F. Weninger, and J. R. Hershey, "Sparse NMF – half-baked or well done?" Mitsubishi Electric Research Labs (MERL), Tech. Rep. TR2015-023, 2015.

[110] E. M. Grais and H. Erdogan, "Single channel speech music separation using nonnegative matrix factorization and spectral masks," in *Proc. ICDSP*, 2011, pp. 1–6.

[111] Z. Ghahramani and M. I. Jordan, "Factorial Hidden Markov Models," *Machine Learning*, vol. 29, no. 2-3, pp. 245–273, 1997.

[112] S. T. Roweis, "One Microphone Source Separation," in *Advances in Neural Information Processing Systems 13*, T. K. Leen, T. G. Dietterich, and V. Tresp, Eds. MIT Press, 2001, pp. 793–799.

[113] A. Ozerov, C. Févotte, and M. Charbit, "Factorial Scaled Hidden Markov Model for polyphonic audio representation and source separation," in *Proc. WASPAA*, 2009, pp. 121–124.

[114] T. Virtanen, "Speech Recognition Using Factorial Hidden Markov Models for Separation in the Feature Space," in *Proc. INTERSPEECH*, 2006, pp. 89–92.

[115] J. R. Hershey and M. Casey, "Audio-Visual Sound Separation Via Hidden Markov Models," in *Proc. NIPS*, 2002, pp. 1173–1180.

[116] M. Stark, M. Wohlmayr, and F. Pernkopf, "Source-Filter-Based Single-Channel Speech Separation Using Pitch Information," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 2, pp. 242–255, 2011.

[117] G. J. Mysore, P. Smaragdis, and B. Raj, "Non-negative Hidden Markov Modeling of Audio with Application to Source Separation," in *Latent Variable Analysis and Signal Separation*, ser. Lecture Notes in Computer Science. Springer, 2010, pp. 140–148.

[118] T. T. Kristjansson, J. R. Hershey, P. A. Olsen, S. J. Rennie, and R. A. Gopinath, "Super-human multi-talker speech recognition: the IBM 2006 speech separation challenge system," in *Proc. INTERSPEECH*, 2006, pp. 97–100.

[119] J. R. Hershey, S. J. Rennie, P. A. Olsen, and T. T. Kristjansson, "Super-human multi-talker speech recognition: A graphical modeling approach," *Computer Speech & Language*, vol. 24, no. 1, pp. 45–66, 2010.

[120] Y.-m. Qian, C. Weng, X.-k. Chang, S. Wang, and D. Yu, "Past review, current progress, and challenges ahead on the cocktail party problem," *Frontiers of Information Technology & Electronic Engineering*, vol. 19, no. 1, pp. 40–63, 2018.

[121] A. W. Rix, J. G. Beerends, M. P. Hollier, and A. P. Hekstra, "Perceptual evaluation of speech quality (PESQ)-a new method for speech quality assessment of telephone networks and codecs," in *Proc. ICASSP*, vol. 2, 2001, pp. 749–752.

[122] "International Telecommunication Union - Recommendation BS.562 : Subjective assessment of sound quality," 1990.

[123] "International Telecommunication Union - Recommendation P.862.1 : Mapping function for transforming P.862 raw result scores to MOS-LQO," 2003.

[124] "International Telecommunication Union - Recommendation P.862.2 : Wideband extension to Recommendation P.862 for the assessment of wideband telephone networks and speech codecs." 2005.

[125] IEEE, "IEEE Recommended Practice for Speech Quality Measurements," *IEEE No 297-1969*, pp. 1–24, 1969.

[126] Y. Hu and P. C. Loizou, "Evaluation of Objective Quality Measures for Speech Enhancement," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 16, no. 1, pp. 229–238, 2008.

[127] C. H. Taal, R. C. Hendriks, R. Heusdens, and J. Jensen, "A short-time objective intelligibility measure for time-frequency weighted noisy speech," in *Proc. ICASSP*, 2010, pp. 4214–4217.

[128] ——, "An Algorithm for Intelligibility Prediction of Time-Frequency Weighted Noisy Speech," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 7, pp. 2125–2136, 2011.

[129] T. M. Elliott and F. E. Theunissen, "The Modulation Transfer Function for Speech Intelligibility," *PLOS Computational Biology*, vol. 5, no. 3, 2009.

[130] R. Drullman, J. M. Festen, and R. Plomp, "Effect of temporal envelope smearing on speech reception," *The Journal of the Acoustical Society of America*, vol. 95, no. 2, pp. 1053–1064, 1994.

[131] S. Jørgensen, J. Cubick, and T. Dau, "Speech Intelligibility Evaluation for Mobile Phones." *Acustica United with Acta Acustica*, vol. 101, pp. 1016–1025, 2015.

[132] T. H. Falk *et al.*, "Objective Quality and Intelligibility Prediction for Users of Assistive Listening Devices: Advantages and limitations of existing tools," *IEEE Signal Processing Magazine*, vol. 32, no. 2, pp. 114–124, 2015.

[133] J. Jensen and C. H. Taal, "Speech Intelligibility Prediction Based on Mutual Information," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 22, no. 2, pp. 430–440, 2014.

[134] R. Xia, J. Li, M. Akagi, and Y. Yan, "Evaluation of objective intelligibility prediction measures for noise-reduced signals in mandarin," in *Proc. ICASSP*, 2012, pp. 4465–4468.

[135] J. Jensen and C. H. Taal, "An Algorithm for Predicting the Intelligibility of Speech Masked by Modulated Noise Maskers," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 11, pp. 2009–2022, 2016.

[136] E. Vincent, H. Sawada, P. Bofill, S. Makino, and J. P. Rosca, "First Stereo Audio Source Separation Evaluation Campaign: Data, Algorithms and Results," in *Independent Component Analysis and Signal Separation*, ser. Lecture Notes in Computer Science. Springer, 2007, pp. 552–559.

[137] V. Emiya, E. Vincent, N. Harlander, and V. Hohmann, "Subjective and Objective Quality Assessment of Audio Source Separation," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 7, pp. 2046–2057, 2011.

[138] E. Vincent, R. Gribonval, and C. Fevotte, "Performance measurement in blind audio source separation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 4, pp. 1462–1469, 2006.

[139] D. Ward, H. Wierstorf, R. Mason, E. Grais, and M. Plumbley, "BSS eval or PEASS? Predicting the perception of singing-voice separation," in *Proc. ICASSP*, 2018, pp. 596 – 600.

[140] E. Cano, D. FitzGerald, and K. Brandenburg, "Evaluation of quality of sound source separation algorithms: Human perception vs quantitative metrics," in *Proc. EUSIPCO*, 2016, pp. 1758–1762.

[141] W. Schultz, P. Dayan, and P. R. Montague, "A Neural Substrate of Prediction and Reward," *Science*, vol. 275, no. 5306, pp. 1593–1599, 1997.

[142] Y. Bengio, "Learning Deep Architectures for AI," *Foundations and Trends in Machine Learning*, vol. 2, no. 1, pp. 1–127, 2009.

[143] L. Deng and D. Yu, "Deep Learning: Methods and Applications," *Foundations and Trends in Signal Processing*, vol. 7, no. 3–4, pp. 197–387, 2014.

[144] T. J. Sejnowski, *The Deep Learning Revolution*. MIT Press, 2018.

[145] R. Lippmann, "An introduction to computing with neural nets," *IEEE ASSP Magazine*, vol. 4, no. 2, pp. 4–22, 1987.

[146] F. Rosenblatt, "The Perceptron: A Probabilistic Model for Information Storage and Organization in The Brain," *Psychological Review*, no. 65, pp. 386–408, 1958.

[147] ——, "Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms." Springer, 1961.

[148] B. Widrow, "An adaptive "ADALINE" neuron using chemical "memistors."." Stanford University, Technical Report 1553-2, 1960.

[149] W. S. McCulloch and W. Pitts, "A logical calculus of the ideas immanent in nervous activity," *The bulletin of mathematical biophysics*, vol. 5, no. 4, pp. 115–133, 1943.

[150] D. O. Hebb, *Organization of Behavior*. Wiley & Sons, 1949.

[151] M. Minsky and S. A. Papert, *Perceptrons: An Introduction to Computational Geometry*. MIT Press, 1969.

[152] M. Olazaran, "A Sociological Study of the Official History of the Perceptrons Controversy," *Social Studies of Science*, vol. 26, no. 3, pp. 611–659, 1996.

[153] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, no. 6088, pp. 533–536, 1986.

[154] J. Schmidhuber, "Deep Learning in Neural Networks: An Overview," *Neural Networks*, vol. 61, pp. 85–117, 2015, arXiv: 1404.7828.

[155] K. Hornik, "Approximation capabilities of multilayer feedforward networks," *Neural Networks*, vol. 4, no. 2, pp. 251–257, 1991.

[156] A. Halevy, P. Norvig, and F. Pereira, "The Unreasonable Effectiveness of Data," *IEEE Intelligent Systems*, vol. 24, no. 2, pp. 8–12, 2009.

[157] N. Shazeer *et al.*, "Outrageously Large Neural Networks: The Sparsely-Gated Mixture-of-Experts Layer," in *Proc. ICLR (arXiv:1701.06538)*, 2017.

[158] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016.

[159] G. E. Hinton and R. R. Salakhutdinov, "Reducing the Dimensionality of Data with Neural Networks," *Science*, vol. 313, no. 5786, pp. 504–507, 2006.

[160] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A Fast Learning Algorithm for Deep Belief Nets," *Neural Computation*, vol. 18, no. 7, pp. 1527–1554, 2006.

[161] Y. Bengio, P. Lamblin, D. Popovici, H. Larochelle, U. D. Montréal, and M. Québec, "Greedy Layer-Wise Training of Deep Networks," in *Advances in Neural Information Processing Systems 19*. MIT Press, 2007, pp. 153–160.

[162] G. E. Hinton, "Training Products of Experts by Minimizing Contrastive Divergence," *Neural Computation*, vol. 14, no. 8, pp. 1771–1800, 2002.

[163] A. Choromanska, M. Henaff, M. Mathieu, G. B. Arous, and Y. LeCun, "The Loss Surfaces of Multilayer Networks," in *Proc. AISTATS (arXiv:1412.0233)*, 2014.

[164] K. Kawaguchi, "Deep Learning without Poor Local Minima," in *Proc. NIPS (arXiv:1605.07110)*, 2016.

[165] G. Hinton *et al.*, "Deep Neural Networks for Acoustic Modeling in Speech Recognition: The Shared Views of Four Research Groups," *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 82–97, 2012.

[166] D. Yu, L. Deng, and G. E. Dahl, "Roles of Pre-Training and Fine-Tuning in Context-Dependent DBN-HMMs for Real-World Speech Recognition," in *Proc. NIPS Deep Learning and Unsupervised Feature Learning Workshop*, 2010.

[167] V. Nair and G. E. Hinton, "Rectified Linear Units Improve Restricted Boltzmann Machines," in *ICML*, 2010, pp. 807–814.

[168] J. Ma, R. P. Sheridan, A. Liaw, G. E. Dahl, and V. Svetnik, "Deep Neural Nets as a Method for Quantitative Structure–Activity Relationships," *Journal of Chemical Information and Modeling*, vol. 55, no. 2, pp. 263–274, 2015.

[169] D. Amodei *et al.*, "Deep Speech 2: End-to-End Speech Recognition in English and Mandarin," *arXiv:1512.02595*, 2015.

[170] W. Xiong *et al.*, "Achieving Human Parity in Conversational Speech Recognition," *arXiv:1610.05256*, 2016.

[171] D. Yu and J. Li, "Recent progresses in deep learning based acoustic models," *IEEE/CAA Journal of Automatica Sinica*, vol. 4, no. 3, pp. 396–409, 2017.

[172] G. Saon *et al.*, "English Conversational Telephone Speech Recognition by Humans and Machines," *arXiv:1703.02136*, 2017.

[173] K. He, X. Zhang, S. Ren, and J. Sun, "Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification," *arXiv:1502.01852*, 2015.

[174] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "DeepFace: Closing the Gap to Human-Level Performance in Face Verification," in *Proc. CVPR*, 2014, pp. 1701–1708.

[175] J. S. Chung, A. Senior, O. Vinyals, and A. Zisserman, "Lip Reading Sentences in the Wild," in *Proc. CVPR*, 2017, pp. 3444–3453.

[176] D. Silver *et al.*, "Mastering the game of Go with deep neural networks and tree search," *Nature*, vol. 529, no. 7587, pp. 484–489, 2016.

[177] ——, "Mastering the game of Go without human knowledge," *Nature*, vol. 550, pp. 354–359, 2017.

[178] M. Moravčík *et al.*, "DeepStack: Expert-level artificial intelligence in heads-up no-limit poker," *Science*, vol. 356, no. 6337, pp. 508–513, 2017.

[179] V. Mnih *et al.*, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, 2015.

[180] R. Miotto, F. Wang, S. Wang, X. Jiang, and J. T. Dudley, "Deep learning for healthcare: review, opportunities and challenges," *Briefings in Bioinformatics*, 2017.

[181] L. Oakden-Rayner, G. Carneiro, T. Bessen, J. C. Nascimento, A. P. Bradley, and L. J. Palmer, "Precision Radiology: Predicting longevity using feature engineering and deep learning methods in a radiomics framework," *Scientific Reports*, vol. 7, no. 1, 2017.

[182] A. Esteva *et al.*, "Dermatologist-level classification of skin cancer with deep neural networks," *Nature*, vol. 542, no. 7639, pp. 115–118, 2017.

[183] N. Wu *et al.*, "Breast density classification with deep convolutional neural networks," *arXiv:1711.03674*, 2017.

[184] Y. Liu *et al.*, "Detecting Cancer Metastases on Gigapixel Pathology Images," *arXiv:1703.02442*, 2017.

[185] D. Bychkov *et al.*, "Deep learning based tissue analysis predicts outcome in colorectal cancer," *Scientific Reports*, vol. 8, no. 1, p. 3395, 2018.

[186] X. Wang *et al.*, "Searching for prostate cancer by fully automated magnetic resonance imaging classification: deep learning versus non-deep learning," *Scientific Reports*, vol. 7, no. 1, p. 15415, 2017.

[187] "Nvidia - Accelerated Computing." [Online]. Available: https://developer.nvidia.com/cuda-zone

[188] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in *Proc. NIPS*, 2012, pp. 1097–1105.

[189] "Viva Technology," 2018. [Online]. Available: https://www.facebook.com/vivatechnologyparis/videos/642529636079722/

[190] K. Hazelwood *et al.*, "Applied Machine Learning at Facebook: A Datacenter Infrastructure Perspective," in *Proc. HPCA*, 2018, pp. 620–629.

[191] PricewaterhouseCoopers, "PwC's Global Artificial Intelligence Study," 2017. [Online]. Available: https://www.pwc.com/gx/en/issues/data-and-analytics/publications/artificial-intelligence-study.html

[192] K. Schwab, "The Fourth Industrial Revolution," *Foreign Affairs*, 2015.

[193] "Andrew Ng: AI Is the New Electricity," 2017. [Online]. Available: https://on.wsj.com/2tc3gLU

[194] R. Rojas, "Networks of width one are universal classifiers," in *Proc. IJCNN*, vol. 4, 2003, pp. 3124–3127.

[195] R. Pascanu, G. Montufar, and Y. Bengio, "On the number of response regions of deep feed forward networks with piece-wise linear activations," in *Proc. ICLR (arXiv:1312.6098)*, 2014.

[196] G. F. Montufar, R. Pascanu, K. Cho, and Y. Bengio, "On the Number of Linear Regions of Deep Neural Networks," in *Proc. NIPS*, 2014, pp. 2924–2932.

[197] G. Huang, Y. Sun, Z. Liu, D. Sedra, and K. Q. Weinberger, "Deep Networks with Stochastic Depth," in *Proc. ECCV*, 2016, pp. 646–661.

[198] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *Proc. CVPR*, 2016, pp. 770–778.

[199] S. Ruder, "An overview of gradient descent optimization algorithms," *arXiv:1609.04747*, 2016.

[200] S. L. Smith, P.-J. Kindermans, C. Ying, and Q. V. Le, "Don't Decay the Learning Rate, Increase the Batch Size," *arXiv:1711.00489*, 2017.

[201] A. C. Wilson, R. Roelofs, M. Stern, N. Srebro, and B. Recht, "The Marginal Value of Adaptive Gradient Methods in Machine Learning," in *Proc. NIPS*, 2017.

[202] X. Glorot, A. Bordes, and Y. Bengio, "Deep Sparse Rectifier Neural Networks," in *Proc. ICAIS*, 2011, pp. 315–323.

[203] M. D. Zeiler *et al.*, "On rectified linear units for speech processing," in *Proc. ICASSP*, 2013, pp. 3517–3521.

[204] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.

[205] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning Internal Representation by Error Propagation," in *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*. MIT Press, 1986, vol. Vol. 1, pp. 318 –362.

[206] R. Pascanu, C. Gulcehre, K. Cho, and Y. Bengio, "How to Construct Deep Recurrent Neural Networks," *arXiv:1312.6026*, 2013.

[207] A. M. Schäfer and H. G. Zimmermann, "Recurrent Neural Networks Are Universal Approximators," in *Artificial Neural Networks – ICANN 2006*, ser. Lecture Notes in Computer Science. Springer, 2006, pp. 632–640.

[208] R. Pascanu, T. Mikolov, and Y. Bengio, "On the difficulty of training Recurrent Neural Networks," in *Proc. ICML (arXiv:1211.5063)*, 2013.

[209] M. Arjovsky, A. Shah, and Y. Bengio, "Unitary Evolution Recurrent Neural Networks," *arXiv:1511.06464*, 2015.

[210] S. Wisdom, T. Powers, J. Hershey, J. Le Roux, and L. Atlas, "Full-Capacity Unitary Recurrent Neural Networks," in *Proc. NIPS*, 2016, pp. 4880–4888.

[211] M. Henaff, A. Szlam, and Y. LeCun, "Orthogonal RNNs and Long-Memory Tasks," in *Proc. ICML (arXiv:1602.06662)*, 2016.

[212] L. Jing *et al.*, "Tunable Efficient Unitary Neural Networks (EUNN) and their application to RNNs," in *Proc. ICML (arXiv:1612.05231)*, 2017.

[213] E. Vorontsov, C. Trabelsi, S. Kadoury, and C. Pal, "On orthogonality and learning recurrent networks with long term dependencies," in *Proc. ICLR (arXiv:1702.00071)*, 2017.

[214] S. Zhang *et al.*, "Architectural Complexity Measures of Recurrent Neural Networks," *arXiv:1602.08210*, 2016.

[215] G.-B. Zhou, J. Wu, C.-L. Zhang, and Z.-H. Zhou, "Minimal gated unit for recurrent neural networks," *International Journal of Automation and Computing*, vol. 13, no. 3, pp. 226–234, 2016.

[216] K. Cho, B. van Merrienboer, D. Bahdanau, and Y. Bengio, "On the Properties of Neural Machine Translation: Encoder-Decoder Approaches," in *Proc. WSSST (arXiv:1409.1259)*, 2014, arXiv: 1409.1259.

[217] K. Cho *et al.*, "Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation," in *Proc. EMNLP (arXiv:1406.1078)*, 2014, arXiv: 1406.1078.

[218] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[219] K. Greff, R. K. Srivastava, J. Koutník, B. R. Steunebrink, and J. Schmidhuber, "LSTM: A Search Space Odyssey," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 10, pp. 2222–2232, 2017.

[220] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling," in *Proc. NIPS Deep Learning and Representation Learning Workshop (arXiv:1412.3555)*, 2014.

[221] J. Collins, J. Sohl-Dickstein, and D. Sussillo, "Capacity and Trainability in Recurrent Neural Networks," *Proc. ICLR (arXiv:1611.09913)*, 2017.

[222] A. Hannun *et al.*, "Deep Speech: Scaling up end-to-end speech recognition," *arXiv:1412.5567*, 2014.

[223] D. Wang and J. Chen, "Supervised Speech Separation Based on Deep Learning: An Overview," *arXiv:1708.07524*, 2017.

[224] Y. Zhang, D. Yu, and G. Chen, "Advanced Recurrent Neural Networks for Automatic Speech Recognition," in *New Era for Robust Speech Recognition*. Springer, 2017, pp. 261–279.

[225] D. Yu and L. Deng, *Automatic Speech Recognition: A Deep Learning Approach*, ser. Signals and Communication Technology. London: Springer London, 2015.

[226] F. Weninger, F. Eyben, and B. Schuller, "Single-channel speech separation with memory-enhanced recurrent neural networks," in *Proc. ICASSP*, 2014, pp. 3709–3713.

[227] F. Weninger *et al.*, "Speech Enhancement with LSTM Recurrent Neural Networks and Its Application to Noise-Robust ASR," in *Proc. LVA/ICA*. Springer, 2015, pp. 91–99.

[228] L. E. Atlas, T. Homma, and R. J. M. II, "An Artificial Neural Network for Spatio-Temporal Bipolar Patterns: Application to Phoneme Classification," in *Proc. NIPS*, 1987, pp. 31–40.

[229] Y. LeCun *et al.*, "Backpropagation Applied to Handwritten Zip Code Recognition," *Neural Computation*, vol. 1, no. 4, pp. 541–551, 1989.

[230] Y. Lecun, "Generalization and network design strategies," Tech. Rep. CRG-TR-89-4, 1989.

[231] R. O. Duda, P. E. Hart, D. G. Stork, C. R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, 2nd ed., 2001.

[232] M. D. Zeiler and R. Fergus, "Visualizing and Understanding Convolutional Networks," in *Proc. ECCV*, ser. Lecture Notes in Computer Science. Springer, 2014, pp. 818–833.

[233] W. Rawat and Z. Wang, "Deep Convolutional Neural Networks for Image Classification: A Comprehensive Review," *Neural Computation*, vol. 29, no. 9, pp. 2352–2449, 2017.

[234] J. Gu *et al.*, "Recent advances in convolutional neural networks," *Pattern Recognition*, vol. 77, pp. 354–377, 2018.

[235] M. Caron, P. Bojanowski, A. Joulin, and M. Douze, "Deep Clustering for Unsupervised Learning of Visual Features," in *Proc. ECCV (arXiv:1807.05520)*, 2018.

[236] C. Hummersone, T. Stokes, and T. Brookes, "On the Ideal Ratio Mask as the Goal of Computational Auditory Scene Analysis," in *Blind Source Separation*, ser. Signals and Communication Technology, G. R. Naik and W. Wang, Eds. Springer, 2014, pp. 349–368.

[237] D. S. Williamson, Y. Wang, and D. Wang, "Complex Ratio Masking for Monaural Speech Separation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 3, pp. 483–492, 2016.

[238] M. Delfarah and D. Wang, "A feature study for masking-based reverberant speech separation," in *Proc. INTERSPEECH*, 2016, pp. 555 – 559.

[239] ——, "Features for Masking-Based Monaural Speech Separation in Reverberant Conditions," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 5, pp. 1085–1094, 2017.

[240] J. Chen, Y. Wang, and D. Wang, "Noise perturbation for supervised speech separation," *Speech Communication*, vol. 78, pp. 1–10, 2016.

[241] J. Chen and D. Wang, "Long Short-Term Memory for Speaker Generalization in Supervised Speech Separation," in *Proc. INTERSPEECH*, 2016, pp. 3314 – 3318.

[242] ——, "Long short-term memory for speaker generalization in supervised speech separation," *The Journal of the Acoustical Society of America*, vol. 141, no. 6, pp. 4705–4714, 2017.

[243] Y. Wang, A. Narayanan, and D. Wang, "On Training Targets for Supervised Speech Separation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 22, no. 12, pp. 1849–1858, 2014.

[244] Y. Wang and D. Wang, "Towards Scaling Up Classification-Based Speech Separation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 7, pp. 1381–1390, 2013.

[245] E. W. Healy, S. E. Yoho, Y. Wang, and D. Wang, "An algorithm to improve speech recognition in noise for hearing-impaired listeners," *The Journal of the Acoustical Society of America*, vol. 134, no. 4, pp. 3029–3038, 2013.

[246] E. W. Healy, S. E. Yoho, Y. Wang, F. Apoux, and D. Wang, "Speech-cue transmission by an algorithm to increase consonant recognition in noise for hearing-impaired listeners," *The Journal of the Acoustical Society of America*, vol. 136, no. 6, pp. 3325–3336, 2014.

[247] A. Narayanan and D. Wang, "Ideal ratio mask estimation using deep neural networks for robust speech recognition," in *Proc. ICASSP*, 2013, pp. 7092–7096.

[248] E. W. Healy, S. E. Yoho, J. Chen, Y. Wang, and D. Wang, "An algorithm to increase speech intelligibility for hearing-impaired listeners in novel segments of the same noise type," *The Journal of the Acoustical Society of America*, vol. 138, no. 3, pp. 1660–1669, 2015.

[249] Y. Wang, J. Chen, and D. Wang, "Deep neural network based supervised speech segregation generalizes to novel noises through large-scale training," Department of Computer Science and Engineering, The Ohio State University, Tech. Rep. OSU-CISRC-3/15-TR02, 2015.

[250] J. Chen, Y. Wang, S. E. Yoho, D. Wang, and E. W. Healy, "Large-scale training to increase speech intelligibility for hearing-impaired listeners in novel noises," *The Journal of the Acoustical Society of America*, vol. 139, no. 5, pp. 2604–2612, 2016.

[251] F. Bolner, T. Goehring, J. Monaghan, B. v. Dijk, J. Wouters, and S. Bleeck, "Speech enhancement based on neural networks applied to cochlear implant coding strategies," in *Proc. ICASSP*, 2016, pp. 6520–6524.

[252] J. J. M. Monaghan *et al.*, "Auditory inspired machine learning techniques can improve speech intelligibility and quality for hearing-impaired listeners," *The Journal of the Acoustical Society of America*, vol. 141, no. 3, pp. 1985–1998, 2017.

[253] T. Goehring, F. Bolner, J. J. M. Monaghan, B. van Dijk, A. Zarowski, and S. Bleeck, "Speech enhancement based on neural networks improves speech intelligibility in noise for cochlear implant users," *Hearing Research*, vol. 344, pp. 183–194, 2017.

[254] Y. H. Lai, F. Chen, S. S. Wang, X. Lu, Y. Tsao, and C. H. Lee, "A Deep Denoising Autoencoder Approach to Improving the Intelligibility of Vocoded Speech in Cochlear Implant Simulation," *IEEE Transactions on Biomedical Engineering*, vol. 64, no. 7, pp. 1568–1578, 2017.

[255] Y.-H. Lai *et al.*, "Deep Learning–Based Noise Reduction Approach to Improve Speech Intelligibility for Cochlear Implant Recipients," *Ear and Hearing*, 2018.

[256] Y. Xu, J. Du, L.-R. Dai, and C.-H. Lee, "An Experimental Study on Speech Enhancement Based on Deep Neural Networks," *IEEE Signal Processing Letters*, vol. 21, no. 1, pp. 65–68, 2014.

[257] D. Liu, P. Smaragdis, and M. Kim, "Experiments on Deep Learning for Speech Denoising," in *Proc. INTERSPEECH*, 2014, pp. 2685 – 2689.

[258] Y. Xu, J. Du, L. R. Dai, and C. H. Lee, "A Regression Approach to Speech Enhancement Based on Deep Neural Networks," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23, no. 1, pp. 7–19, 2015.

[259] F. Weninger, J. R. Hershey, J. L. Roux, and B. Schuller, "Discriminatively trained recurrent neural networks for single-channel speech separation," in *GlobalSIP*, 2014, pp. 577–581.

[260] H. Erdogan, J. R. Hershey, S. Watanabe, and J. L. Roux, "Phase-sensitive and recognition-boosted speech separation using deep recurrent neural networks," in *Proc. ICASSP*, 2015, pp. 708–712.

[261] ——, "Deep Recurrent Networks for Separation and Recognition of Single-Channel Speech in Nonstationary Background Audio," in *New Era for Robust Speech Recognition*. Springer, 2017, pp. 165–186.

[262] D. S. Williamson and D. Wang, "Time-Frequency Masking in the Complex Domain for Speech Dereverberation and Denoising," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 7, pp. 1492–1501, 2017.

[263] J. Du, Y. Tu, Y. Xu, L. Dai, and C. H. Lee, "Speech separation of a target speaker based on deep neural networks," in *ICSP*, 2014, pp. 473–477.

[264] P. S. Huang, M. Kim, M. Hasegawa-Johnson, and P. Smaragdis, "Deep learning for monaural speech separation," in *Proc. ICASSP*, 2014, pp. 1562–1566.

[265] P.-S. Huang, M. Kim, M. Hasegawa-Johnson, and P. Smaragdis, "Joint Optimization of Masks and Deep Recurrent Neural Networks for Monaural Source Separation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23, no. 12, pp. 2136–2147, 2015.

[266] C. Weng, D. Yu, M. L. Seltzer, and J. Droppo, "Deep Neural Networks for Single-Channel Multi-Talker Speech Recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23, no. 10, pp. 1670–1679, 2015.

[267] Y. Wang, J. Du, L. R. Dai, and C. H. Lee, "Unsupervised single-channel speech separation via deep neural network for different gender mixtures," in *Proc. APSIPA*, 2016, pp. 1–4.

[268] X. L. Zhang and D. Wang, "A Deep Ensemble Learning Method for Monaural Speech Separation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 5, pp. 967–977, 2016.

[269] E. W. Healy, M. Delfarah, J. L. Vasko, B. L. Carter, and D. Wang, "An algorithm to increase intelligibility for hearing-impaired listeners in the presence of a competing talker," *The Journal of the Acoustical Society of America*, vol. 141, no. 6, pp. 4230–4239, 2017.

[270] Y.-H. Tu, J. Du, and C.-H. Lee, "A Speaker-Dependent Approach to Single-Channel Joint Speech Separation and Acoustic Modeling Based on Deep Neural Networks for Robust Recognition of Multi-Talker Speech," *Journal of Signal Processing Systems*, pp. 1–11, 2017.

[271] S. Venkataramani, J. Casebeer, and P. Smaragdis, "End-to-end Source Separation with Adaptive Front-Ends," in *Proc. NIPS Machine Learning for Audio Signal Processing Workshop*, 2017.

[272] M. Delfarah and D. Wang, "Recurrent neural networks for cochannel speech separation in reverberant environments," in *Proc. ICASSP*, 2018, pp. 5404–5408.

[273] L. Bramsløw, G. Naithani, A. Hafez, T. Barker, N. H. Pontoppidan, and T. Virtanen, "Improving competing voices segregation for hearing impaired listeners using a low-latency deep neural network algorithm," *The Journal of the Acoustical Society of America*, vol. 144, no. 1, pp. 172–185, 2018.

[274] J. R. Hershey, Z. Chen, J. L. Roux, and S. Watanabe, "Deep clustering: Discriminative embeddings for segmentation and separation," in *Proc. ICASSP*, 2016, pp. 31–35.

[275] J. MacQueen, "Some methods for classification and analysis of multivariate observations." The Regents of the University of California, 1967.

[276] Y. Isik, J. L. Roux, Z. Chen, S. Watanabe, and J. R. Hershey, "Single-Channel Multi-Speaker Separation Using Deep Clustering," in *Proc. INTERSPEECH*, 2016, pp. 545–549.

[277] Y. Luo, Z. Chen, J. R. Hershey, J. L. Roux, and N. Mesgarani, "Deep clustering and conventional networks for music separation: Stronger together," in *Proc. ICASSP*, 2017, pp. 61–65.

[278] Z.-Q. Wang, J. L. Roux, and J. R. Hershey, "Alternative Objective Functions for Deep Clustering," in *Proc. ICASSP*, 2018, pp. 686–690.

[279] Z.-Q. Wang, J. L. Roux, D. Wang, and J. R. Hershey, "End-to-End Speech Separation with Unfolded Iterative Phase Reconstruction," in *Proc. INTERSPEECH*, 2018.

[280] Z. Chen, Y. Luo, and N. Mesgarani, "Deep attractor network for single-microphone speaker separation," in *Proc. ICASSP*, 2017, pp. 246–250.

[281] Y. Luo, Z. Chen, and N. Mesgarani, "Speaker-Independent Speech Separation With Deep Attractor Network," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 4, pp. 787–796, 2018.

[282] C. Stephenson, P. Callier, A. Ganesh, and K. Ni, "Monaural speaker separation using source-contrastive estimation," in *Proc. SiPS*, 2017, pp. 1–6.

[283] M. Bianchini and F. Scarselli, "On the Complexity of Neural Network Classifiers: A Comparison Between Shallow and Deep Architectures," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 25, no. 8, pp. 1553–1565, 2014.

[284] S. Gupta, A. Agrawal, K. Gopalakrishnan, and P. Narayanan, "Deep Learning with Limited Numerical Precision," in *Proc. ICML (arXiv:1502.02551)*, 2015.

[285] M. Tu, V. Berisha, Y. Cao, and J.-s. Seo, "Reducing the Model Order of Deep Neural Networks Using Information Theory," *arXiv:1605.04859*, 2016.

[286] J. Garland and D. Gregg, "Low Complexity Multiply Accumulate Unit for Weight-Sharing Convolutional Neural Networks," *arXiv:1609.05132*, 2016.

[287] M. Kim and P. Smaragdis, "Bitwise Neural Networks for Efficient Single-Channel Source Separation," in *Proc. NIPS Machine Learning for Audio Signal Processing Workshop*, 2017.

[288] P. Molchanov, S. Tyree, T. Karras, T. Aila, and J. Kautz, "Pruning Convolutional Neural Networks for Resource Efficient Inference," in *Proc. ICLR*, 2017.

[289] R. J. Cintra, S. Duffner, C. Garcia, and A. Leite, "Low-Complexity Approximate Convolutional Neural Networks," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–12, 2018.

[290] Y. Hoshen, R. J. Weiss, and K. W. Wilson, "Speech acoustic modeling from raw multichannel waveforms," in *Proc. ICASSP*, 2015, pp. 4624–4628.

[291] H. Erdogan, J. R. Hershey, S. Watanabe, M. I. Mandel, and J. L. Roux, "Improved MVDR Beamforming Using Single-Channel Mask Prediction Networks," in *Proc. INTERSPEECH*, 2016, pp. 1981–1985.

[292] J. Heymann, L. Drude, and R. Haeb-Umbach, "Neural network based spectral mask estimation for acoustic beamforming," in *Proc. ICASSP*, 2016, pp. 196–200.

[293] X. Zhang and D. Wang, "Deep Learning Based Binaural Speech Separation in Reverberant Environments," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 5, pp. 1075–1084, 2017.

[294] L. Drude and R. Haeb-Umbach, "Tight Integration of Spatial and Spectral Features for BSS with Deep Clustering Embeddings," in *Proc. INTERSPEECH*. ISCA, 2017, pp. 2650–2654.

[295] C. Boeddeker, H. Erdogan, T. Yoshioka, and R. Haeb-Umbach, "Exploring Practical Aspects of Neural Mask-Based Beamforming for Far-Field Speech Recognition," in *Proc. ICASSP*, 2018, pp. 6697–6701.

[296] J. Heymann, L. Drude, and R. Haeb-Umbach, "A generic neural acoustic beamforming architecture for robust multi-channel speech processing," *Computer Speech & Language*, vol. 46, pp. 374–385, 2017.

[297] J. Heymann, L. Drude, C. Boeddeker, P. Hanebrink, and R. Haeb-Umbach, "Beamnet: End-to-end training of a beamformer-supported multi-channel ASR system," in *Proc. ICASSP*, 2017, pp. 5325–5329.

[298] C. Boeddeker, P. Hanebrink, L. Drude, J. Heymann, and R. Haeb-Umbach, "Optimizing neural-network supported acoustic beamforming by algorithmic differentiation," in *Proc. ICASSP*, 2017, pp. 171–175.

[299] Z.-Q. Wang, J. L. Roux, and J. R. Hershey, "Multi-Channel Deep Clustering : Discriminative Spectral and Spatial Embeddings for Speaker-Independent Speech Separation," in *Proc. ICASSP*, 2018, pp. 1–5.

[300] J. Heymann, M. Bacchiani, and T. N. Sainath, "Performance of Mask based Statistical Beamforming in a Smart Home Scenario," in *Proc. ICASSP*, 2018, pp. 6722–6726.

[301] H. Mcgurk and J. Macdonald, "Hearing lips and seeing voices," *Nature*, vol. 264, no. 5588, pp. 746–748, 1976.

[302] W. H. Sumby and I. Pollack, "Visual Contribution to Speech Intelligibility in Noise," *The Journal of the Acoustical Society of America*, vol. 26, no. 2, pp. 212–215, 1954.

[303] A. Owens and A. A. Efros, "Audio-Visual Scene Analysis with Self-Supervised Multisensory Features," *arXiv:1804.03641*, 2018.

[304] A. Gabbay, A. Ephrat, T. Halperin, and S. Peleg, "Seeing Through Noise: Visually Driven Speaker Separation and Enhancement," 2018, pp. 3051–3055.

[305] A. Ephrat *et al.*, "Looking to Listen at the Cocktail Party: A Speaker-Independent Audio-Visual Model for Speech Separation," in *Proc. SIGGRAPH (arXiv:1804.03619)*, 2018.

[306] Y. Zhao, B. Xu, R. Giri, and T. Zhang, "Perceptually Guided Speech Enhancement using Deep Neural Networks," in *Proc. ICASSP*, 2018, pp. 5074–5078.

[307] H. Zhang, X. Zhang, and G. Gao, "Training Supervised Speech Separation System to Improve STOI and PESQ Directly," in *Proc. ICASSP*, 2018, pp. 5374–5378.

[308] S. W. Fu, T. W. Wang, Y. Tsao, X. Lu, and H. Kawai, "End-to-End Waveform Utterance Enhancement for Direct Evaluation Metrics Optimization by Fully Convolutional Neural Networks," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 9, pp. 570 – 1584, 2018.

[309] G. Naithani, J. Nikunen, L. Bramsløw, and T. Virtanen, "Deep neural network based speech separation optimizing an objective estimator of intelligibility for low latency applications," in *Proc. IWAENC*, 2018.

[310] A. H. Andersen, J. M. d. Haan, Z. H. Tan, and J. Jensen, "Predicting the Intelligibility of Noisy and Nonlinearly Processed Binaural Speech," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 11, pp. 1908–1920, 2016.

[311] Y. Koizumi, K. Niwa, Y. Hioka, K. Kobayashi, and Y. Haneda, "DNN-based source enhancement self-optimized by reinforcement learning using sound quality measurements," in *Proc. ICASSP*, 2017, pp. 81–85.

[312] R. Fakoor, X. He, I. Tashev, and S. Zarar, "Reinforcement Learning To Adapt Speech Enhancement to Instantaneous Input Signal Quality," in *Proc. NIPS Machine Learning for Audio Signal Processing Workshop*, 2017.

[313] S. R. Park and J. Lee, "A Fully Convolutional Neural Network for Speech Enhancement," in *Proc. INTERSPEECH*, 2017, arXiv: 1609.07132.

[314] S. W. Fu, Y. Tsao, X. Lu, and H. Kawai, "Raw waveform-based speech enhancement by fully convolutional networks," in *Proc. APSIPA*, 2017, pp. 6–12.

[315] S. Bai, J. Z. Kolter, and V. Koltun, "Convolutional Sequence Modeling Revisited," in *Proc. ICLR*, 2018.

[316] N. Zeghidour, N. Usunier, G. Synnaeve, R. Collobert, and E. Dupoux, "End-to-End Speech Recognition From the Raw Waveform," in *Proc. INTERSPEECH*, 2018.

[317] J. Lee, T. Kim, J. Park, and J. Nam, "Raw Waveform-based Audio Classification Using Sample-level CNN Architectures," in *Proc. NIPS Machine Learning for Audio Signal Processing Workshop*, 2017.

This page intentionally left blank.

Part II

Papers

This page intentionally left blank.

Paper A

Speech Intelligibility Potential of General and Specialized Deep Neural Network Based Speech Enhancement Systems

Morten Kolbæk, Zheng-Hua Tan, and Jesper Jensen

The paper has been published in
IEEE/ACM Transactions on Audio, Speech, and Language Processing,
vol. 25, no. 1, pp. 153–167, January 2017.

© 2016 IEEE
The layout has been revised.

Abstract

In this paper we study aspects of single microphone Speech Enhancement (SE) based on Deep Neural Networks (DNNs). Specifically, we explore the generalizability capabilities of state-of-the-art DNN based SE systems with respect to the background noise type, the gender of the target speaker, and the Signal-to-Noise Ratio (SNR). Furthermore, we investigate how specialized DNN based SE systems, which have been trained to be either noise type specific, speaker specific or SNR specific, perform relative to DNN based SE systems that have been trained to be noise type general, speaker general and SNR general. Finally, we compare how a DNN based SE system trained to be noise type general, speaker general and SNR general performs relative to a state-of-the-art Short-Time Spectral Amplitude Minimum Mean Square Error (STSA-MMSE) based SE algorithm.

We show that DNN based SE systems, when trained specifically to handle certain speakers, noise types and SNRs, are capable of achieving large improvements in estimated Speech Quality (SQ) and Speech Intelligibility (SI), when tested in matched conditions. Furthermore, we show that improvements in estimated SQ and SI can be achieved by a DNN based SE system when exposed to unseen speakers, genders and noise types, given a large number of speakers and noise types have been used in the training of the system. In addition, we show that a DNN based SE system that has been trained using a large number of speakers and a wide range of noise types outperforms a state-of-the-art STSA-MMSE based SE method, when tested using a range of unseen speakers and noise types. Finally, a listening test using several DNN based SE systems tested in unseen speaker conditions show that these systems can improve SI for some SNR and noise type configurations but degrade SI for others.

1 Introduction

Improving quality and intelligibility of noisy speech signals is of great interest in a vast amount of applications such as mobile communications, speech recognition systems, and hearing aids. In a single-microphone setting, improving Speech Quality (SQ) and especially Speech Intelligibility (SI) is a challenging task and is an active topic of research [1–3]. Traditionally, single microphone Speech Enhancement (SE) has been addressed by statistical model based methods such as the Wiener filter [2] and Short-Time Spectral Amplitude Minimum Mean Square Error (STSA-MMSE) estimators, e.g., [4–6]. However, recent advances in Computational Auditory Scene Analysis (CASA) and machine learning have introduced new methods, e.g. Deep Neural Network (DNN), Gaussian Mixture Model (GMM), and Support Vector Machine (SVM) based methods, which address single-microphone SE and speech segregation in terms of advanced statistical estimators. These estimators aim at estimating either a clean speech Time-Frequency (T-F) represen-

tation directly or a T-F mask that is applied to the T-F representation of the noisy speech to arrive at an estimate of the clean speech signal [7–15]. For some potential future applications, e.g. DNN based SE algorithms for hearing aids or mobile communications, the range of possible acoustic situations which can realistically occur is virtually endless. It is therefore important to understand how such methods perform in different acoustic situations, and how they perform, when they are exposed to "unseen" situations, i.e. acoustic scenarios not encountered during training. Despite the obvious importance of this generalizability question, it is currently not well understood.

In this study we focus on situations where a single target speaker is present in additive noise and the aim of the SE algorithm is to enhance the speech signal and attenuate the noise using a single-microphone recording. Generally, when evaluating generalizability of machine learning based SE algorithms, there are at least three dimensions in which the input signal can vary: i) the noise type dimension, ii) the speaker dimension and iii) the Signal to Noise Ratio (SNR) dimension. Therefore, evaluation of DNN based SE methods should cover each of these dimensions in a way similar to what is expected to be experienced in a real life scenario. For mobile communication devices and hearing aid systems, evaluation should hence encompass a wide range of noise types, a wide range of speakers and a wide range of SNRs, in order to give a realistic estimate of the expected performance of the algorithm in real life scenarios. On the other hand, for applications where the typical usage situation is much more well-defined, e.g. voice-controlled devices to be used in a car cabin situation, training and testing might involve only car cabin noises at a narrow SNR range for a single particular speaker.

The exploration of these three dimensions is motivated by the fact that no matter how many noise types, SNRs, and speakers a SE system is exposed to during training, in a real life scenario, sooner or later the system will be exposed to an unseen noise type, an unseen speaker or an unseen SNR. However, if the system is well trained, one might expect that the system has captured some general acoustic characteristics from these dimensions and hence generalizes well to unseen conditions. Furthermore, if any *a priori* knowledge about the noise type, speaker characteristics or SNR is available, it is important to know what performance gain can be achieved by utilizing this *a priori* knowledge.

Several studies have investigated aspects of generalizability of SE algorithms based on DNNs, SVMs, and GMMs, e.g. [7–24]. However, these models are fundamentally different in both training schemes and architectures [25] and since DNNs are currently state-of-the-art in a large number of applications [26] and have outperformed SVMs and GMMs in SE tasks [8–10, 12], SVMs and GMMs are less suited for the current study and are therefore not considered.

Common for all the studies, based on DNNs [7, 9, 12–14, 16–20], is that during training or testing one or more of the generalizability dimensions defined above are held fixed, while others are varied. To the authors knowledge no study exists which explores the simultaneous variation of all the three dimensions - a situation which is realistic for many real life applications. Furthermore, interpretation of existing studies is sometimes complicated by the fact that the training and test signals, for the dimensions which *are* varied, are not described in all details. For example, the distribution of males and females is often not reported [7, 16, 18, 19] and it is hence not clear if the system is mostly a gender specific or gender general system. Several studies [7, 16, 18] use the TIMIT corpus [27], which is approximately 70% male and 30% female.Furthermore, the duration of the different training noise types is typically not considered when the training data is constructed, hence the exact distribution of the noise types is unknown. For example, in [8, 16, 18, 19] noise sequences with highly varying duration are used, which makes it unclear to which extent these systems are noise specific or noise general. Another issue related to the noise dimension is concerning the construction of training and test data. In several studies [8, 10–12, 18, 19] , the exact same noise realizations were used for training and testing. In [28] this training/testing paradigm was analyzed, and it was found to erroneously give remarkably better performance compared to the realistic scenario, where the actual noise sequence is unknown. Furthermore, the systems presented in [7, 9, 12–14, 16–20] are based on various network architectures, training methods, testing methods, speech corpora, noise databases, feature representations, target representations etc. As a consequence of these differences, their results cannot be directly related and it is therefore unclear how a state-of-the-art DNN based SE algorithm perform when the generalizability dimensions mentioned above are considered simultaneously. Finally, it is unclear to what extent state-of-the-art DNN based SE algorithms provide improvements over existing non-DNN based SE methods. In [7, 16, 29] a DNN based SE method similar to the one studied here outperforms several different non-DNN based methods such as statistical MMSE based methods [6, 30–33] and non-Negative Matrix Factorization (NMF) methods [7, 34]. However, since the DNNs used in [7, 16, 29] have not been trained across all three generalizability dimensions, the comparison may not yield a true picture of the actual performance difference. This is particularly true with the statistical MMSE based methods [1, 2], which have not been trained to handle any specific noise types or speakers but merely rely on basic statistical assumptions with respect to Short-Time Fourier Transform (STFT) coefficients and might perform worse than a system trained on a given speaker or noise type.

The goal of this paper is therefore to conduct a systematic evaluation of the generalizability capabilities of a state-of-the-art DNN based SE algorithm in terms of estimated SQ and SI. Specifically, we investigate how a state-of-

the-art DNN based SE method performs when it is trained to be noise type specific vs. noise type general, speaker specific vs. speaker general, and SNR specific vs. SNR general. Furthermore, we study the performance drop, if any, for systems which are specialized in one or more of the three generalizability dimensions, compared to a completely general DNN based SE system, which relies on essentially no prior knowledge with respect to speaker characteristics, noise type, and SNR. Additionally, it is investigated how this general system performs compared to a state-of-the-art non-machine learning based method namely the STSA-MMSE estimator employing generalized gamma priors as proposed in [6, 32, 33][1]. This is of interest since the STSA-MMSE method relies on very little prior knowledge compared to conventional DNN based SE methods [8, 9, 13]. Furthermore, given that the computational and memory complexity associated with DNN type of systems is typically orders of magnitude larger than that associated with simple STSA-MMSE based systems it is of obvious interest to understand the performance gain of this technology. Finally, a listening test is conducted, using both specialized and general DNN based SE systems, to investigate if such systems improve SI, when tested in different matched and unmatched conditions.

It is important to note that this paper emphasizes on the generalizability properties of DNN based SE algorithms in terms of estimated and measured SI, since these properties has not yet been rigorously investigated in the current literature [7, 9, 12–14, 16–20]. To do so, we rely on a specific implementation of a feed-forward DNN, whose architecture and training procedure resemble those of a large range of existing DNN based SE methods [7, 9, 16, 19, 35]. This allows us to expect that our findings are representative not only for our particular implementation but are generally valid for DNN based SE methods. The fact that the DNN based SE method under study is a representative member of a larger class of algorithms also implies that this particular implementation does not necessarily outperform all existing methods with respect to estimated SQ and SI.

Furthermore, obviously, the three chosen generalizability dimensions are not the only dimensions for which mixing scenarios can vary. Other such dimensions include reverberation conditions, e.g. in terms of varying room impulse responses, or digital signal processing conditions, e.g. in terms of signal sampling rate, number of bits with which each sample is represented, microphone characteristics, compression/coding schemes, etc. Furthermore, for DNN based SE algorithms the DNN architecture can also be varied and considered as a dimension. We have chosen the speaker dimension, the noise type dimension and the SNR dimension for this particular work since these are dimensions most often encountered in the SE literature [7–24]. Furthermore, in most papers related to DNN based SE algorithms only a single

[1] http://insy.ewi.tudelft.nl/content/software-and-data

speaker is considered, so it is of interest to study how well these algorithms generalize to unknown speakers. Finally, the performance of non-machine learning based SE algorithms such as STSA-MMSE and Wiener filtering based approaches are known to be highly dependent on the noise type, and SNR, but not the speaker. Hence, it is of interest to study how a DNN based SE algorithm performs in a large range of noise types, speakers and SNRs.

The paper is organized as follows: Section 2 describes the DNN architecture, training procedure and speech material used for conducting the desired experiments. Section 3 describes and discusses the experimental setups and results. Finally, in Section 4 the findings are concluded.

2 Speech Enhancement Using Neural Networks

2.1 Speech Corpus and Noisy Mixtures

The phonetically balanced Danish speech corpus *Akustiske Databaser for Dansk (ADFD)*[2] is used as target speech material for training and testing all DNN based SE systems considered in this paper. This corpus consists of two sets: One set (set 1) consisting of 56 speakers with 986 spoken utterances for each speaker and another set (set 2) with 560 speakers and 311 spoken utterances, and males and females are approximately equally distributed among the two sets. The majority of the text material is based on conversational speech, but also single words, numbers and sequences of numbers are included, and each utterance has an average duration of approximately 5 seconds.

The training, validation, and test sets, were constructed such that no sentence appears more than once in the combined training, validation, and test set. The sampling frequency was 16 kHz and all files were normalized to have unit Root Mean Square (RMS) power.

The noisy mixtures for all experiments were constructed by adding a noise signal to a clean speech signal at a certain SNR. The noise signal was scaled to achieve the desired SNR based on the speech active region of the speech signal, i.e. the silence parts in the beginning and in the end of the speech signal were omitted in SNR computation. Omitting the silence parts for the SNR computation is crucial since the inclusion of these parts will effectively decrease the energy estimate of the clean speech, hence a lower noise power is required to achieve the same SNR, than if the silence regions were omitted. The difference in SNR between these two approaches of constructing noisy mixtures can be more than one dB and is typically not considered in the literature [9, 16, 18], even though it is of importance if results from different studies are to be related. Alternatively, a Voice Activity Detection (VAD) algorithm could have been used to exclude all silent regions, which would

[2] https://www.nb.no/sbfil/dok/nst_taledat_no.pdf

be highly beneficial for practical applications. However, for simplicity and to be in-line with existing literature [7–24], we excluded the VAD for all experiments. As before, the global SNR based approach were chosen from a practical perspective and to be in-line with existing literature [7–24], where global SNR is by far the most common.

2.2 Features and Labels

The choice of training targets for supervised speech enhancement have been widely studied [2, 7, 36–41]. Recent studies [7, 9, 38, 40] suggest that continuous targets such as the Ideal Ratio Mask (IRM) are preferable over binary targets such as the Ideal Binary Mask (IBM) [39, 40]. Therefore, the DNN studied in this paper is trained in a supervised fashion to estimate the IRM from a feature representation of a noisy speech signal.

The T-F representation used to construct the IRM is based on a gammatone filter bank with 64 filters linearly spaced on a MEL frequency scale from 50 Hz to 8 kHz and with a bandwidth equal to one Equivalent Rectangular Bandwidth (ERB) [42][3]. The output of the filter bank is divided into 20 ms frames with 10 ms overlap and with a sampling frequency of 16 kHz, each T-F unit represents a vector of 320 samples.

Let $\mathbf{x}(n,\omega)$ denote the Time-Frequency (T-F) unit of the clean speech signal at frame n and frequency channel ω, and let $\mathbf{d}(n,\omega)$ denote the corresponding T-F unit of the noise signal. Then the IRM is computed as [7]

$$\mathrm{IRM}(n,\omega) = \left(\frac{||\mathbf{x}(n,\omega)||^2}{||\mathbf{x}(n,\omega)||^2 + ||\mathbf{d}(n,\omega)||^2} \right)^{\beta},$$

where $||\mathbf{x}(n,\omega)||^2$ is the squared 2-norm, i.e. the clean speech energy, of T-F unit n in frequency channel ω. Likewise, $||\mathbf{d}(n,\omega)||^2$ is the noise energy of a T-F unit n in frequency channel ω. The variable β is a tunable parameter and has for all experiments in this paper been set to $\beta = 0.5$, which in [7] was found empirically to provide good results.

To have discriminative and noise robust features, each frame is transformed into a 1845-dimensional feature vector inspired by [3, 8, 9, 12, 43–45]. Although, very recent works use only magnitude spectra [13, 20, 46] a large context of several hundred milliseconds is used, which is undesirable for real time applications. The chosen feature vector was found to outperform features of magnitude spectra when these were based on only a small context. The features used are 31 Mel-Frequency Cepstrum Coefficient (MFCC), 15 Amplitude Modulation Spectrogram (AMS), 13 Relative Spectral Transform - Perceptual Linear Prediction (RASTA-PLP) and 64 Gammatone Filter bank Energies (GFE). Furthermore delta and double delta features are computed

[3] http://web.cse.ohio-state.edu/pnl/shareware/cochleagram

and a context of 2 past and 2 future frames are utilized, hence arriving at a 1845-dimensional feature vector. All feature vectors are normalized to zero mean and unit variance.

2.3 Network Architecture and Training

The DNNs used in this paper follow a feed-forward architecture with a 1845-dimensional input layer and three hidden layers, each with 1024 hidden units, and 64 output units [7, 9]. The activation functions for the hidden units are Rectified Linear Units (ReLUs) [47] and for the output units the sigmoid function is applied. The hidden layers are initialized using the "GlorotUniform" approach [48]. Furthermore, the DNN has approximately 4 million tunable parameters in terms of weights and biases. The values of the parameters are found using Stochastic Gradient Descent (SGD) following the AdaGrad approach [49]. The gradients are computed using backpropagation based on the Mean Square Error (MSE) error function using a batch size of 1024 [25]. Furthermore, 20% dropout has been applied to all hidden layers during training to reduce overfitting [50]. In order to further reduce overfitting, an early-stopping training scheme is applied, which stops the training, when the MSE of the validation set has not decreased with more than 1% for more than 20 epochs. Although used in [8, 16], unsupervised DNN pre-training using Deep Belief Networks [26, 51, 52] was found not to significantly improve performance and has therefore not been applied in the reported results.

Finally, it is well known that increasing the network size or changing the network architecture can improve performance of DNN based algorithms [13, 16, 20, 46, 53, 54]. However, it is not practically feasible to include network architecture as a dimension in our experiments. Furthermore, although absolute performance might be better with a different architecture, the conclusions drawn from using a fixed-sized feed-forward DNN are expected to be valid for a broader range of DNN architectures, since the underlying assumptions are practically the same.

2.4 Signal Enhancement

After DNN training, the IRM is estimated for a given test signal by forward propagating its feature representation, for all frames, through the DNN. The output of the DNN represents the estimated IRM, $\widehat{\mathrm{IRM}}(n,\omega)$ for the given frame. The estimated IRM can then be applied to the T-F representation of the noisy speech signal by multiplying the given entry of the mask to all 320 noisy signal samples of a T-F unit. All T-F units in each frequency channel are then concatenated and all overlapping parts are summed. Afterwards, the 64 frequency channels can be synthesized into a time domain signal by first compensating for the different group delays in the different channels and

then adding the frequency channels. The group delay compensation is performed by time reversing the signals, passing them through the gammatone filter bank and then time reversing the signals once again [42].

2.5 Evaluation of Enhancement Performance

Speech signals enhanced with the DNN based SE algorithm studied in this paper were evaluated using the Short-Time Objective Intelligibility (STOI) [55] measure and the wideband extension of the Perceptual Evaluation of Speech Quality (PESQ) measure [56, 57]. The STOI measure estimates SI and PESQ estimates SQ and both have been found to be highly correlated with human listening tests [2, 55]. STOI is defined in the range $[-1, 1]$ and PESQ is defined in the range $[1, 4.5]$ and for both measures higher is better. We used the implementations of STOI and PESQ available from [55] and [2], respectively.

Although other performance measures exists such as Signal to Distortion Ratio (SDR), Signal to Interferences Ratio (SIR), and Signal to Artifact Ratio (SAR)[2, 58] we report only PESQ and STOI to limit the number of tables. Furthermore, PESQ and STOI are by far the most used speech quality and speech intelligibility estimators in the literature [7–24].

3 Experimental Results and Discussion

To investigate the generalizability capability of DNN based SE systems with relation to: i) the noise type dimension, ii) the speaker dimension and iii) the SNR dimension, three experimental setups, one for each dimension, have been designed. When a dimension is explored the remaining two dimensions are held fixed. For example, when exploring the SNR dimension, the SNR dimension is varied but only a single speaker and a single noise type is used for both training and testing. Furthermore, a fourth setup has been constructed where a general system has been designed. This system was trained using a wide range of speakers, noise types and SNRs, hence the system relies on a minimum of *a priori* knowledge. This "general" system is compared against the three experiments previously described, as well as a state-of-the-art non-machine learning based SE algorithm.

3.1 SNR Dimension

The purpose of the SNR experiments is to investigate the impact on the performance of DNN based SE systems, when training is performed based on a single SNR vs. a wide range of SNRs, i.e. constructing a SNR specific or a SNR general system. The SNR dimension is explored using speech material based on 986 spoken utterances from a single female speaker from the

ADFD set 1. These 986 utterances were divided such that 686 were used for training, 100 for validation and 200 for testing. Two noise types have been investigated, a stationary Speech Shaped Noise (SSN) and a non-stationary Babble (BBL) noise. The SSN sequence is constructed by filtering a 50 min. Gaussian white noise sequence through a 12th-order all-pole filter with coefficients found from Linear Predictive Coding (LPC) analysis of 100 randomly chosen TIMIT sentences [27]. The BBL noise is also based on TIMIT. The corpus, which consists of a total of 6300 spoken sentences, is randomly divided into 6 groups of 1050 concatenated utterances. Each group is then truncated to equal length followed by addition of the six groups. This results in a BBL noise sequence with a duration of over 50 min. The SSN and BBL sequences were both divided such that 40 min. were used for training, 5 min. were used for validation and 5 min. for testing, hence there is no overlapping samples in the noise segments used for training, validation and test. To investigate how the performance of DNN based SE systems depends on the SNR dimension, eight systems were trained with eight different SNR settings for both SSN and BBL noise. All 16 systems were tested using eight SNRs ranging from -15 dB to 20 dB with steps of 5 dB. For each noise source, the first system was trained using -5 dB since this is a commonly encountered SNR in the literature [8, 9] where SI is typically degraded and DNN based SE algorithms have been successfully applied [8, 9]. The next system was trained using SNRs from -5 dB to 0 dB with steps of 1 dB. In a similar fashion wider and wider SNR ranges were used for training the remaining systems with the widest range being from -15 dB to 20 dB. The precise intervals are given in Tables A.1, A.2, A.3 and A.4. For all systems, each training utterance was mixed with different noise realizations 35 times in order to increase the amount of training data. For each noisy mixture, the SNR was drawn from a discrete uniform distribution defined within the given SNR range. Due to the large number of realizations, it is assumed that the distribution of drawn SNRs is approximately uniform. The noise signal used for each noisy mixture was extracted from the whole training noise sequence by using a starting index drawn from a discrete uniform distribution defined over the entire length of the noise sequence. If the starting index is such that there is no room for the whole utterance, the remaining samples are extracted from the beginning of the noise sequence. Following the same procedure, each validation utterance is mixed with different noise realizations 10 times. Using this form of training data augmentation, the total amount of training utterances, used for training each system, is increased to $686 \times 35 = 24010$, which is approximately equal to 33 hours of speech material and is approximately 65% more data than used by [9].

The results of the SNR dimension experiments are presented in Tables A.1 and A.2 for SSN and in Tables A.3 and A.4 for BBL noise. From Tables A.1 and A.3 it is seen that the SNR specific system of SNR of -5 dB achieves

relatively large STOI improvements, for test signal SNRs in the range -10 dB to 5 dB. In general, it can be observed that inclusion of SNRs in the range from -15 dB to 5 dB has a larger positive impact on the performance than inclusion of SNRs above 5 dB. This might be explained partly by the fact that intelligibility is almost 100% (STOI $\approx$ 1) for test signal SNRs above 5 dB, and partly by the limited noise energy, which makes it more difficult for the DNN to actually learn important noise characteristics. Tables A.2 and A.4 show a somewhat similar picture. The inclusion of training signals with SNRs around 0 dB in general improves performance, but extending the training SNR range from 5 dB to 20 dB does not further improve performance. Furthermore, it is also seen that the system in general cannot improve PESQ for test signals with SNRs below -5 dB.

Based on these experiments it can be concluded that there is generally a good correspondence between SNR ranges used in training and STOI improvement seen during testing. For example, the systems trained in the SNR range from -5 dB to 0 dB perform better at 0 dB than the systems trained using only -5 dB. Furthermore, even at -15 dB, where the noise energy is approximately 40 times larger than the speech energy, STOI is still improved with 0.074 and 0.093 for SSN and BBL noise, respectively, when this particular SNR is included in the training set, and the improvement is almost constant for SSN, and even slightly increasing for BBL noise, when a wider range of positive SNRs are included in the training set. Also, the system trained using the widest SNR range from -15 dB to 20 dB achieves almost similar performance as the -5 dB SNR specialized system, when tested at an SNR of -5 dB, and generally performs better at other SNRs. This observation is in line with related studies [59] and is of large practical importance, as it suggests that DNN based SE systems should simply be trained using as large a training signal SNR range as practically possible.

Table A.1: STOI improvement for the SNR dimension. Eight DNN based SE systems trained on different SNR ranges as indicated in the first row. The noise type dimension is held constant using SSN only and the speaker dimension is held constant using a single female speaker. The systems are evaluated using STOI for test signals with 8 different SNRs ranging from -15dB to 20dB. The second column presents the STOI score for the unprocessed noisy mixtures. Columns 3-10 present STOI improvements.

	Noisy	–5dB	–5dB – 0dB	–5dB – 5dB	–10dB – 5dB	–15dB – 5dB	–15dB – 10dB	–15dB – 15dB	–15dB – 20dB
-15dB	0.354	0.016	0.019	0.028	0.063	0.074	0.075	0.075	0.072
-10dB	0.417	0.170	0.166	0.165	0.186	0.186	0.185	0.183	0.179
-5dB	0.519	**0.219**	**0.218**	**0.218**	**0.219**	**0.216**	**0.215**	**0.213**	**0.210**
0dB	0.642	0.180	0.186	0.187	0.185	0.183	0.183	0.182	0.181
5dB	0.756	0.115	0.125	0.130	0.128	0.126	0.128	0.127	0.127
10dB	0.844	0.058	0.070	0.078	0.077	0.076	0.079	0.079	0.078
15dB	0.905	0.016	0.030	0.040	0.039	0.039	0.044	0.045	0.044
20dB	0.944	-0.010	0.005	0.015	0.014	0.014	0.020	0.023	0.023

Table A.2: As Table A.1 but for PESQ.

	Noisy	–5dB	–5dB – 0dB	–5dB – 5dB	–10dB – 5dB	–15dB – 5dB	–15dB – 10dB	–15dB – 15dB	–15dB – 20dB
-15dB	1.133	-0.044	-0.041	-0.044	-0.036	-0.027	-0.029	-0.035	-0.032
-10dB	1.115	0.025	0.024	0.025	0.038	0.044	0.042	0.041	0.041
-5dB	1.115	0.198	0.190	0.192	0.202	0.196	0.196	0.191	0.187
0dB	1.144	0.457	0.425	0.421	0.410	0.400	0.410	0.408	0.405
5dB	1.234	0.700	0.691	0.655	0.643	0.638	0.630	0.636	0.642
10dB	1.438	**0.769**	**0.875**	0.879	0.863	0.859	0.831	0.803	0.811
15dB	1.811	0.583	0.830	**0.942**	**0.925**	**0.911**	**0.948**	**0.902**	**0.878**
20dB	2.346	0.130	0.518	0.764	0.745	0.733	0.860	0.877	0.848

Table A.3: As Table A.1 but for BBL.

	Noisy	–5dB	–5dB – 0dB	–5dB – 5dB	–10dB – 5dB	–15dB – 5dB	–15dB – 10dB	–15dB – 15dB	–15dB – 20dB
-15dB	0.292	0.048	0.034	0.033	0.070	0.093	0.095	0.094	0.096
-10dB	0.369	0.161	0.150	0.146	0.170	0.173	0.173	0.174	0.174
-5dB	0.480	**0.214**	**0.218**	**0.216**	**0.214**	**0.205**	**0.205**	**0.206**	**0.206**
0dB	0.608	0.187	0.200	0.202	0.194	0.188	0.189	0.191	0.191
5dB	0.728	0.128	0.147	0.152	0.146	0.141	0.144	0.147	0.146
10dB	0.823	0.070	0.091	0.098	0.095	0.091	0.096	0.098	0.097
15dB	0.890	0.024	0.045	0.056	0.053	0.050	0.057	0.059	0.059
20dB	0.934	-0.008	0.013	0.026	0.023	0.021	0.029	0.032	0.033

Table A.4: As Table A.1 but for PESQ and BBL.

	Noisy	–5dB	–5dB – 0dB	–5dB – 5dB	–10dB – 5dB	–15dB – 5dB	–15dB – 10dB	–15dB – 15dB	–15dB – 20dB
-15dB	1.201	-0.063	-0.066	-0.058	-0.070	-0.080	-0.066	-0.079	-0.075
-10dB	1.180	-0.047	-0.060	-0.058	-0.056	-0.052	-0.055	-0.055	-0.054
-5dB	1.143	0.086	0.090	0.089	0.081	0.074	0.072	0.079	0.075
0dB	1.162	0.289	0.312	0.319	0.294	0.280	0.279	0.284	0.293
5dB	1.270	0.493	0.571	0.580	0.543	0.511	0.516	0.527	0.538
10dB	1.478	**0.636**	0.772	0.805	0.770	0.732	0.740	0.745	0.741
15dB	1.829	0.621	**0.826**	**0.914**	**0.884**	**0.844**	**0.872**	**0.872**	**0.854**
20dB	2.312	0.426	0.691	0.863	0.835	0.794	0.863	0.871	0.851

3.2 Noise Dimension

The purpose of the noise dimension experiments is to investigate the performance impact when DNN based SE systems are trained on a single noise type vs. a wide range of noise types. In other words, this allows us to compare a noise specific vs. a noise general system. The noise dimension has been explored using the same 986 spoken utterances from the same single female speaker as used in the SNR experiments. Likewise, the partition of the speech material into training, validation and test set is also identical to the SNR experiments. To explore the noise dimension, six distinct noise types were used: SSN (N1) and BBL (N2) from the SNR experiments and four additional noises: street (N3), pedestrian (N4), cafe (N5) and bus (N6), from the CHiME3 dataset [60]. Furthermore, 1260 randomly selected sound effect noises from soundbible.com[4] were used to construct a seventh noise type referred to as the *mix* (N7) noise type. These 1260 noises were first truncated

[4]http://soundbible.com/free-sound-effects-1.html

to have a maximum duration of 3 seconds each and then concatenated into one large noise sequence. The sound effects include sounds from animals, singing humans, explosions, airplanes, slamming doors etc. All seven (N1 – N7) noise types used for the noise experiments were first truncated to have a total duration of 50 min. and then divided into a 40 min. training set, a 5 min. validation set and a 5 min. test set, hence there is no overlapping samples in the noise segments used for training, validation and test.

To investigate how the performance of the DNN based SE system depends on the noise dimension, eight systems were trained with eight different noise combinations all at an SNR of -5 dB. Two systems were trained with only one noise type, namely the stationary SSN (N1) and non-stationary BBL (N2). The remaining six systems were trained with an increasing number of noise types starting with N1 – N2 and ending with N1 – N7 as indicated in the second row in Tables A.5 and A.6. When noise types were combined, the 40 min. noise sequences were concatenated and similar to the SNR experiment, a noise sequence was extracted based on a randomly chosen starting index within this concatenated noise sequence. Similarly to the SNR experiment, each utterance in the training set was mixed with a randomly chosen noise sequence 35 times, hence a total of $686 \times 35 = 24010$ noise mixtures were constructed. The large number of mixtures and the identical duration of the noise sequences ensures that the noise distribution within the training data is approximately uniform, hence a noise-general system is constructed. All eight systems have been tested with speech signals contaminated by all seven noise types, which ensures that all but the system trained with all seven noises will be tested with at least one unseen and at the most 6 unseen noise types.

The results are presented in Tables A.5 and A.6 where the first column represents the noise types used for testing and the second row represents the noises used for training. Table A.5 shows that when a system is trained using SSN only (N1) it achieves a relatively large STOI improvement of 0.22, when tested on that particular noise type, but generalizes poorly on the majority of the unseen test noises. Similarly, when a system is trained on BBL (N2), the performance is good in the matched noise case, but the system generalizes poorly to other noise types. Furthermore, when both SSN and BBL noise types are included equally in the training set (N1-N2), the system performs almost as good as the individual noise specific systems. However, the system does not generalize as well to the unseen noises as N1 did alone, except for the *mix* noise type, that similarly to BBL is highly non-stationary. It is also interesting to notice that the SSN and BBL specific systems achieve very similar performance for test signals contaminated by SSN and BBL, respectively. This is in contrast to STFT-based methods for which non-stationary noise is much more challenging [1]. A different picture is seen when a third noise (N3) is added in the training set (N1-N3). This system performs similarly

well in the matched noise type setting, but also for the unseen noises the performance has increased considerably. Similar behavior is seen when the remaining noise types are included in the training set. Furthermore, even though new noise types are included in the training set, the performance of the system is almost constant in the matched noise type setting. One can argue that *str*, *ped*, *caf* and *bus* are quite similar noise types, but it is seen that the system trained with signals contaminated by all but the mix noise type (N1-N6) generalizes relatively well to the mix noise type, which is a noise type radically different from the others. From Table A.6 a similar behavior is observed where relatively large PESQ scores are achieved for all testing noises, already after noise type N1 – N3 have been included in the training set. Similar for both Tables A.5 and A.6 is that there is generally a good correspondence between noise types used for training and STOI and PESQ improvements seen during testing. For example, the systems performing best on SSN and BBL noise are the systems that have been trained on only these noise types. However, a system trained on both noise types show only a slightly decrease in performance. Furthermore, the noise general system (N1 – N7), where all seven noise types are used for training, achieves on average the best performance across all seven noise types, while still being comparable in performance to the more specialized systems where only a single or a few noise types have been used for training. This is similar to the SNR experiments where no particular degradation in performance was observed by extending the SNR range used for training.

Table A.5: STOI improvement for the noise type dimension. Eight DNN based SE systems have been trained with different combinations of seven different noise types (N1-N7) as given by the first row. The SNR dimension is held constant at -5dB and the speaker dimension is held constant using a single female speaker. The systems have been evaluated using STOI and test signals corrupted by all seven noise types. The second column presents the STOI score for the noisy unprocessed mixtures. Columns 3-10 present STOI improvements.

	Noisy	N1	N2	N1–N2	N1-N3	N1–N4	N1–N5	N1–N6	N1–N7
N1: ssn	0.519	**0.220**	0.083	0.207	0.209	**0.208**	**0.206**	**0.206**	**0.203**
N2: bbl	0.482	0.029	**0.217**	**0.210**	**0.211**	0.204	0.202	0.203	0.199
N3: str	0.590	0.122	-0.079	0.080	0.174	0.172	0.172	0.173	0.171
N4: ped	0.504	0.095	-0.008	0.078	0.139	0.157	0.161	0.160	0.158
N5: caf	0.572	0.072	-0.007	0.065	0.143	0.155	0.165	0.167	0.165
N6: bus	0.703	0.071	-0.058	0.003	0.112	0.114	0.118	0.130	0.128
N7: mix	0.685	0.015	0.028	0.038	0.072	0.078	0.092	0.093	0.119

3.3 Speaker Dimension

The purpose of the speaker dimension experiments is to study the impact of using a single speaker vs. a wide range of speakers in the training material, i.e. constructing a speaker specific or speaker general system.

Table A.6: As Table A.5 but for PESQ.

	Noisy	N1	N2	N1–N2	N1-N3	N1–N4	N1–N5	N1–N6	N1–N7
N1: ssn	1.112	**0.197**	-0.012	**0.175**	0.186	0.174	0.175	0.178	0.173
N2: bbl	1.174	-0.072	0.060	0.048	0.054	0.047	0.039	0.046	0.032
N3: str	1.069	0.114	-0.002	0.071	0.302	0.294	0.294	0.294	0.298
N4: ped	1.099	0.033	-0.025	0.005	0.095	0.118	0.125	0.125	0.120
N5: caf	1.081	0.030	-0.003	0.025	0.191	0.224	0.237	0.247	0.242
N6: bus	1.083	0.125	0.010	0.036	**0.329**	**0.336**	**0.351**	**0.421**	**0.415**
N7: mix	1.143	0.002	**0.067**	0.059	0.126	0.144	0.161	0.180	0.293

The speaker dimension is explored using speech material based on 311 spoken utterances from 41 male and 41 female speakers from the ADFD set 2. The utterances from the 41 females are referenced as F-ID1 – F-ID41 and similarly, the utterances from the 41 males are referenced as M-ID1 – M-ID41. For each of the speakers of interest, (F/M-ID1 – F/M-ID21) 231 utterances were used for training, 30 for validation and 50 for testing. Furthermore, 50 utterances from each of the 40 remaining speakers (F/M-ID22 – F/M-ID41) were used as testing material for unseen speaker testing. The text material used for the 50 test utterances from each speaker was the same for all 82 speakers used for these experiments.

A total of 10 systems were trained. Five systems using speech material corrupted with SSN at an SNR of -5 dB and five systems with speech material corrupted with BBL noise at an SNR of -5 dB. For each noise type, speakers F-ID1 and MID-1 were used to train two individual speaker specific systems. Furthermore, speakers F-ID2 – 21 and M-ID2 – 21 were used to train two individual gender specific systems and finally the speakers F-ID2 – 21 and M-ID2 – 21 were combined (F/M-ID2 – 21) and used to train a single speaker general system.

All systems were evaluated in both a seen speaker and an unseen speaker scenario using the test material from speaker F/M-ID1 – F/M-ID21 and F/M-ID22 – F/M-ID41, respectively. However, the systems trained using only one speaker is tested using 20 speakers, instead of only one speaker, to give more realistic unseen-speaker results. Since the number of distinct utterances used for training vary between the different systems, due to the varying number of speakers, a fixed total number of 18480 training utterances were used for training all systems. This is done to ensure that all systems are presented to the same amount of noise material. Using the same argument a total number of 1200 utterances were used for validation during the training of all systems. To achieve 18480 training mixtures, and 1200 validation mixtures, for each system, each distinct utterance was mixed with unique noise realizations multiple times as given by Table A.7.

The results with SSN are presented in Tables A.8 and A.9, and the results with BBL noise are presented in Tables A.10 and A.11. The first col-

Table A.7: Training and validation data augmentation scheme used for results reported in subsection 3.3 to ensure all systems use the same amount of data. The format is the following: $\#speakers \times \#utterances \times \#repetitions = \#mixtures$

System	#Training Utterances	#Validation Utterances
Speaker Specific	$1 \times 231 \times 80 = 18480$	$1 \times 30 \times 40 = 1200$
Gender Specific	$20 \times 231 \times 4 = 18480$	$20 \times 30 \times 2 = 1200$
Speaker General	$40 \times 231 \times 2 = 18480$	$40 \times 30 \times 1 = 1200$

umn presents the speaker IDs used for testing and the second row represents speaker IDs used for training.

From Table A.8 it is seen that speaker specific systems trained on a single speaker achieves a STOI improvement of 0.168 and 0.204 for same-gender-same-speaker testing, for the female (F-ID1) and male (M-ID1) specific systems, respectively. However, if these systems are tested with new speakers of same gender, i.e. same-gender-new-speaker testing, the STOI improvements are reduced to 0.127 and 0.114 for the female (F-ID1) and male (M-ID1) specific systems, respectively. Furthermore, if the systems are tested on opposite gender the STOI improvement decreases to 0.067 and 0.062 for the female (F-ID1) and male (M-ID1) specific systems, respectively. Similar behavior, but with larger variations, is seen from Table A.10 where the systems have been trained using utterances corrupted with BBL noise instead of SSN. Table A.10 shows that systems trained using F-ID1 and M-ID2 improve STOI with 0.131 and 0.184 for same-gender-same-speaker testing, for the female (F-ID1) and male (M-ID1) systems, respectively. However, these improvements reduce to 0.046 and 0.039 for same-gender-new-speakers testing, and to -0.093 and -0.107 for new-gender testing, for the female (F-ID1) and male (M-ID1) systems, respectively.

From these results it can be concluded that systems which are trained using only a single speaker generalizes very well to unseen utterances from the same speaker but not as good to unseen utterances from new speakers of same gender and even worse to opposite gender. Especially for BBL noise, the systems even degrade the signals when evaluated using opposite gender.

If gender specific systems are trained with 20 speakers instead of only a single speaker it is seen from Table A.8 that the STOI improvements in the same-gender-same-speakers testing case are 0.175 and 0.174 for the female (F-ID2 – F-ID21) and male (M-ID2 – M-ID21) systems, respectively. Furthermore, the STOI improvements in the same-gender-new-speakers testing case are 0.170 and 0.160 for the female (F-ID2 – F-ID21) and male (M-ID2 – M-ID21) systems, respectively. Compared to the systems trained using a single speaker, the systems trained using 20 speakers of same gender generalize considerably better to the same-gender-new-speaker testing case. Also in the

new-gender testing case Table A.8 shows STOI improvements of 0.124 and 0.119 for the female (F-ID2 – F-ID21) and male (M-ID2 – M-ID21) systems, respectively. However, Table A.10 shows that STOI is degraded when the female (F-ID2 – F-ID21) and male (M-ID2 – M-ID21) systems are tested in the new-gender testing case. Finally, if a speaker general system is trained using both males and females (F/M-ID2 – 21) and is tested in an unseen speaker setting based on both genders (F/M-ID22 – 41) using respectively SSN and BBL noise, the STOI improvements are 0.164 and 0.111, respectively. This shows that the speaker general system in terms of STOI generalizes considerably better than the speaker specific and gender specific systems to unseen speakers of both genders, for both a stationary and non-stationary noise type. Importantly, it is seen that the loss from a gender specific system to a gender general system is almost zero.

One interesting observation is the decrease in performance, when compared to the experiments exploring the noise type dimension in subsection 3.2. For example, Table A.5 shows that a system specialized to a single female speaker using BBL noise at an SNR of -5 dB achieves a STOI improvement of 0.217. Table A.10 shows that a similar system (F-ID1) trained with a different female speaker using BBL noise at an SNR of -5 dB achieves a STOI improvement of 0.131, which is a considerable difference. There is one major difference between these two systems. For the experiments used to produce Table A.5, the speaker was represented by 686 distinct spoken utterances, whereas for the experiments used to produce Table A.10 only 231 distinct spoken utterances were used. This indicates that not only the number of speakers but also the variability in speech material from each speaker is crucial to achieve good generalizability.

In general, it can be observed that a DNN based SE system trained using a single speaker becomes speaker specific and performs well, in terms of estimated SI, when evaluated using the same speaker. If a large number of speakers, of the same gender, are used for training, the system becomes gender specific and generalizes well to unseen speakers of same gender. Furthermore, if a large number of male and female speakers are used for training, the system becomes speaker general and generalizes well to unseen speakers of both genders. This applies for systems trained using training signals corrupted with either SSN or BBL noise. In terms of estimated SQ a similar behavior can only be observed for systems trained with training signals corrupted with SSN whereas for the systems trained using training signals corrupted with BBL noise no, or only minor, improvements were found as shown by Tables A.9 and A.11.

Table A.8: STOI improvement for the speaker dimension. Five DNN based SE systems have been trained on a varying number of speakers of both genders as given by the first row. The systems have been tested in both speaker matched and unmatched conditions. The noise type dimension is held constant using SSN for training and testing and the SNR dimension is held constant using an SNR of -5dB for training and testing. The systems have been evaluated using STOI. The second column presents the STOI score for the noisy unprocessed mixtures. Columns 3-7 present STOI improvements

Test \ Train	Noisy	F-ID1	M-ID1	F-ID2 – 21	M-ID2 – 21	F/M-ID2 – 21
F-ID1	0.564	**0.168**	–	–	–	–
M-ID1	0.460	–	**0.204**	–	–	–
F-ID2-21	0.532	–	–	**0.175**	–	**0.170**
F-ID22-41	0.530	0.127	0.062	0.170	0.119	0.166
M-ID2-21	0.543	–	–	–	**0.174**	0.167
M-ID22-41	0.538	0.067	0.114	0.124	0.160	0.163
F/M-ID2–21	0.538	–	–	–	–	0.167
F/M-ID22–41	0.535	0.097	0.089	0.147	0.140	0.164

Table A.9: As Table A.8 but for PESQ

Test \ Train	Noisy	F-ID1	M-ID1	F-ID2 – 21	M-ID2 – 21	F/M-ID2 – 21
F-ID1	1.062	**0.160**	–	–	–	–
M-ID1	1.078	–	**0.185**	–	–	–
F-ID2-21	1.068	–	–	**0.168**	–	0.158
F-ID22-41	1.065	0.108	0.043	0.160	0.058	0.150
M-ID2-21	1.110	–	–	–	**0.219**	0.208
M-ID22-41	1.118	0.070	0.149	0.141	0.199	**0.213**
F/M-ID2–21	1.096	–	–	–	–	0.175
F/M-ID22–41	1.093	0.087	0.095	0.149	0.126	0.180

Table A.10: As Table A.8 but for BBL

Test \ Train	Noisy	F-ID1	M-ID1	F-ID2 – 21	M-ID2 – 21	F/M-ID2 – 21
F-ID1	0.535	**0.131**	–	–	–	–
M-ID1	0.433	–	**0.184**	–	–	–
F-ID2-21	0.498	–	–	**0.121**	–	0.110
F-ID22-41	0.496	0.046	-0.107	0.117	-0.059	0.108
M-ID2-21	0.511	–	–	–	**0.140**	0.112
M-ID22-41	0.507	-0.093	0.039	-0.007	0.125	**0.115**
F/M-ID2–21	0.505	–	–	–	–	0.110
F/M-ID22–41	0.501	-0.025	-0.034	0.054	0.032	0.111

3.4 Combined Dimensions

The purpose of the combined dimension experiments is twofold. First, we wish to determine the performance decrease, if any, of a general DNN based SE system vs. the specialized systems considered in the three previous subsections, where only one dimension was varied at a time. Such experiments can be used to relate results previously reported in the literature, where at least one dimension has been fixed, to the more general case where all three dimensions are varied. Secondly, we wish to investigate how such a general

Table A.11: As Table A.8 but for PESQ and BBL

Test \ Train	Noisy	F-ID1	M-ID1	F-ID2 – 21	M-ID2 – 21	F/M-ID2 – 21
F-ID1	1.094	**0.065**	–	–	–	–
M-ID1	1.159	–	**0.029**	–	–	–
F-ID2-21	1.129	–	–	**0.001**	–	-0.007
F-ID22-41	1.130	-0.029	-0.049	-0.007	-0.064	-0.017
M-ID2-21	1.168	–	–	–	**0.024**	-0.005
M-ID22-41	1.181	-0.074	-0.040	-0.077	0.002	-0.007
F/M-ID2–21	1.141	–	–	–	–	**0.001**
F/M-ID22–41	1.164	-0.059	-0.051	-0.050	-0.037	-0.019

DNN based SE method performs relative to a state-of-the-art non-machine learning based method, namely the STSA-MMSE method proposed in [6]. This is done in an attempt to give a realistic picture of the performance difference between these two classes of algorithms, which utilize different kinds of prior knowledge.

Alternatively, we could have compared the performance with a NMF based SE approach, which is another popular SE algorithm. However, several studies [3, 61–63] show that DNN based SE algorithms outperform NMF based approaches on several tasks. Furthermore, the NMF based approach can be viewed as a single hidden layer DNN. Hence, comparing the performance of the DNN based SE algorithm investigated in this paper to a NMF based SE algorithm is less interesting than a comparison with the STSA-MMSE based SE approach, which is from a completely different class of algorithms.

STSA-MMSE type of methods such as [6, 64] are very general and make only few assumptions about the target and noise signals and are therefore often used in practice [1]. Furthermore, the performance of these simple non-machine learning based algorithms in terms of speech intelligibility improvements are well studied in the literature, e.g. [37, 40, 65, 66]. Although deep neural network based speech enhancement algorithms have shown impressive performance, they are often trained and tested in narrow settings using either a few noise types [9, 43] or a single speaker [13]. It is therefore of interest to identify if/when a deep neural network based speech enhancement algorithm can outperform a non-machine learning based method, when approximately the same type of general a priori information is utilized: given that the computational and memory complexity associated with deep neural network type of systems is typically orders of magnitude larger than that associated with simple STSA-MMSE based systems, it is of obvious interest to understand the performance gain one can expect from the increased memory and computational complexity.

The comparison is based on a "General" DNN based SE system trained using all the noise types from the noise dimension experiments at all the SNRs

from the SNR dimension experiments, and using all the speakers from the speaker dimension experiments. This means that the system is trained using 7 different and equally distributed noise types mixed with 20 female and 20 male speakers at SNRs from -15 dB to 20 dB. To encompass the increased variability of this dataset compared to the previous datasets the training set size is increased to $40 \times 231 \times 12 = 110880$ utterances. To make a fair comparison to the STSA-MMSE method, which does not strongly rely on prior speaker, SNR, or noise type knowledge, 10 unseen noises, 20 unseen females and 20 unseen males are used for evaluating the performance at SNRs from -10 dB to 20 dB. The 10 noises are taken from the DEMAND noise database[5] and represent a wide range of both stationary and non-stationary noise types.

The STSA-MMSE method relies on the assumption that noise free Discrete Fourier Transform (DFT) coefficients are distributed according to a generalized gamma distribution with parameters $\gamma = 2$ and $\nu = 0.15$ [1, 6]. The *a priori* SNR estimator used by the STSA-MMSE method is the Decision-Directed approach [64] using a smoothing factor of 0.98 and a noise Power Spectral Density (PSD) estimate based on the noise PSD tracker reported in [67][6]. For each utterance, the noise tracker was initialized using a noise PSD estimate based on a noise only region prior to speech activity.

The results of the experiments are presented in Tables A.12 and A.14 for the STSA-MMSE method and in Tables A.13 and A.15 for the general DNN based SE system. The performance scores for the noisy unprocessed mixtures are given in parenthesis and the average across all 10 noises at each SNR is given in the last row. From Tables A.12 and A.13 it is seen that for all SNRs, the DNN based SE system outperforms the STSA-MMSE method in terms of STOI. Similar behavior is seen from Tables A.14 and A.15, where the systems are evaluated using PESQ. However, at high SNRs the STSA-MMSE method achieves comparable results with the DNN based SE method and for some noise types such as *DM station* and *DM traffic*, the STSA-MMSE even achieves slightly better PESQ scores at SNRs above 5 dB. This might be explained by the fact that the STSA-MMSE algorithm uses prior knowledge in terms of an ideal noise PSD estimate based on a noise only signal region prior to speech activity. This prior knowledge could be particularly beneficial for stationary noise types, where the initial noise PSD estimate remains correct throughout the utterance. The DNN based SE method explored in this paper does not utilize such prior knowledge. However, in [16, 68] noise PSD estimates obtained prior to speech activity were used in combination with traditional features to train a DNN based SE system and it was shown that performance was improved, when such prior knowledge was utilized. It is also seen that the STSA-MMSE method on average does not improve STOI, whereas the gen-

[5] http://parole.loria.fr/DEMAND

[6] http://insy.ewi.tudelft.nl/content/software-and-data

eral DNN based SE method does. For some conditions such as *DM station* at an SNR of -5 dB the improvement is as high as 0.096. In general, it can be observed that a DNN based SE system trained across all three generalizability dimensions using a large number of noise types, speakers and SNRs, outperforms a state-of-the-art non-machine learning based method, even though this method utilizes prior knowledge in terms of ideal initial noise PSD estimates. However, the performance of the general DNN based SE system is on average considerably reduced compared to the specialized systems where only one generalizability dimension was varied at a time. From this, it can be concluded that if the usage situation of a SE algorithm is well-defined e.g., in terms of speaker characteristics, noise type, or SNR range, considerably performance improvements can be achieved using a DNN based SE algorithm that has been specifically trained to fit the application. On the other hand, for more general applications where the acoustic usage situation cannot be narrowed down in one or more of these dimensions, the advantage of DNN based SE methods is much smaller, while they may still offer improvements over current state-of-the-art non-machine learning based methods.

Table A.12: Average STOI performance improvement scores using a state-of-the-art STSA-MMSE estimator. The score in the parenthesis is for the noisy unprocessed signals. The test material is based on 2000 utterances evenly distributed among 20 males and 20 females mixed with 10 different noise types from the DEMAND noise corpus at seven SNRs in the range from -10 dB to 20 dB.

	-10dB	-5dB	0dB	5dB	10dB	15dB	20dB
DM bus	-0.003 (0.819)	-0.003 (0.884)	-0.003 (0.927)	-0.003 (0.955)	-0.003 (0.972)	-0.003 (0.983)	-0.003 (0.991)
DM cafe	-0.026 (0.521)	-0.011 (0.643)	-0.003 (0.756)	-0.001 (0.843)	-0.002 (0.902)	-0.003 (0.939)	-0.003 (0.962)
DM cafeter	-0.043 (0.459)	-0.022 (0.58)	-0.006 (0.706)	-0.001 (0.811)	-0.002 (0.884)	-0.003 (0.928)	-0.003 (0.955)
DM car	0.008 (0.913)	0.005 (0.945)	0.002 (0.966)	0.000 (0.979)	-0.001 (0.987)	-0.002 (0.992)	-0.002 (0.996)
DM metro	0.002 (0.62)	0.007 (0.73)	0.006 (0.821)	0.002 (0.886)	0.000 (0.929)	-0.001 (0.955)	-0.002 (0.972)
DM resto	-0.054 (0.395)	-0.031 (0.496)	-0.012 (0.623)	-0.004 (0.746)	-0.003 (0.84)	-0.004 (0.902)	-0.004 (0.939)
DM river	0.011 (0.55)	0.020 (0.655)	0.020 (0.755)	0.013 (0.838)	0.006 (0.897)	**0.002** (0.936)	**0.000** (0.961)
DM square	-0.008 (0.651)	-0.001 (0.761)	-0.000 (0.846)	-0.002 (0.904)	-0.003 (0.94)	-0.003 (0.962)	-0.003 (0.977)
DM station	0.008 (0.496)	**0.022** (0.614)	**0.023** (0.733)	**0.016** (0.829)	**0.008** (0.894)	**0.002** (0.934)	**0.000** (0.958)
DM traffic	**0.019** (0.611)	0.021 (0.724)	0.016 (0.819)	0.009 (0.887)	0.003 (0.93)	0.001 (0.957)	-0.001 (0.974)
Average	-0.009 (0.604)	0.001 (0.703)	0.004 (0.795)	0.003 (0.868)	0.000 (0.917)	-0.001 (0.949)	-0.002 (0.968)

3.5 Listening Test

To investigate how the DNN based SE system performs in practice, an intelligibility test, using 10 normal-hearing Danish graduate students, has been conducted. The gender distribution among the 10 students was 3 females and 7 males with ages from 20 to 28 years and a mean age of 24. Five systems have been designed for the SI test and their training specifications are given by Table A.16. The systems are designed to investigate if a female specific system, in different noise and SNR conditions (DNN-1 – DNN-4), can

Table A.13: As Table A.12 but for a state-of-the-art DNN based SE algorithm.

	-10dB	-5dB	0dB	5dB	10dB	15dB	20dB
DM bus	0.033 (0.819)	0.021 (0.884)	0.011 (0.927)	0.004 (0.955)	-0.001 (0.972)	-0.004 (0.983)	-0.006 (0.991)
DM cafe	0.034 (0.521)	0.058 (0.643)	0.054 (0.756)	0.038 (0.843)	0.022 (0.902)	0.010 (0.939)	0.002 (0.962)
DM cafeter	-0.011 (0.459)	0.038 (0.58)	0.056 (0.706)	0.045 (0.811)	0.027 (0.884)	0.014 (0.928)	0.005 (0.955)
DM car	0.018 (0.913)	0.008 (0.945)	0.002 (0.966)	-0.002 (0.979)	-0.004 (0.987)	-0.006 (0.992)	-0.007 (0.996)
DM metro	0.040 (0.62)	0.046 (0.73)	0.038 (0.821)	0.024 (0.886)	0.012 (0.929)	0.004 (0.955)	-0.002 (0.972)
DM resto	-0.017 (0.395)	0.046 (0.496)	0.078 (0.623)	**0.069** (0.746)	**0.043** (0.84)	**0.022** (0.902)	**0.009** (0.939)
DM river	0.077 (0.55)	0.089 (0.655)	0.074 (0.755)	0.048 (0.838)	0.026 (0.897)	0.011 (0.936)	0.001 (0.961)
DM square	0.064 (0.651)	0.054 (0.761)	0.036 (0.846)	0.021 (0.904)	0.010 (0.94)	0.003 (0.962)	-0.003 (0.977)
DM station	0.076 (0.496)	**0.096** (0.614)	**0.080** (0.733)	0.051 (0.829)	0.027 (0.894)	0.013 (0.934)	0.004 (0.958)
DM traffic	**0.085** (0.611)	0.072 (0.724)	0.048 (0.819)	0.027 (0.887)	0.013 (0.93)	0.004 (0.957)	-0.002 (0.974)
Average	0.040 (0.604)	0.053 (0.703)	0.048 (0.795)	0.032 (0.868)	0.018 (0.917)	0.007 (0.949)	0.000 (0.968)

Table A.14: As Table A.12 but for PESQ.

	-10dB	-5dB	0dB	5dB	10dB	15dB	20dB
DM bus	0.172 (1.26)	0.278 (1.51)	0.325 (1.93)	0.336 (2.46)	0.264 (3.05)	0.129 (3.61)	-0.001 (4.04)
DM cafe	-0.035 (1.13)	0.013 (1.11)	0.067 (1.20)	0.142 (1.42)	0.206 (1.83)	0.222 (2.37)	0.174 (2.97)
DM cafeter	-0.061 (1.17)	-0.001 (1.11)	0.056 (1.16)	0.129 (1.32)	0.206 (1.65)	0.228 (2.14)	0.177 (2.73)
DM car	**0.363** (1.21)	**0.500** (1.45)	**0.682** (1.81)	**0.742** (2.35)	0.677 (2.94)	0.459 (3.53)	0.192 (4.01)
DM metro	0.019 (1.13)	0.134 (1.17)	0.264 (1.33)	0.356 (1.64)	0.370 (2.12)	0.297 (2.70)	0.180 (3.28)
DM resto	-0.130 (1.25)	-0.046 (1.13)	0.029 (1.11)	0.113 (1.20)	0.227 (1.43)	0.305 (1.84)	0.295 (2.40)
DM river	0.009 (1.07)	0.061 (1.09)	0.183 (1.17)	0.410 (1.36)	0.605 (1.75)	0.632 (2.30)	0.500 (2.92)
DM square	0.039 (1.08)	0.102 (1.14)	0.203 (1.29)	0.303 (1.61)	0.344 (2.10)	0.330 (2.68)	0.247 (3.29)
DM station	-0.008 (1.09)	0.093 (1.07)	0.271 (1.13)	0.511 (1.30)	0.688 (1.63)	**0.705** (2.14)	**0.565** (2.75)
DM traffic	0.054 (1.07)	0.188 (1.09)	0.406 (1.19)	0.615 (1.43)	**0.706** (1.86)	0.700 (2.43)	0.558 (3.05)
Average	0.042 (1.14)	0.132 (1.19)	0.249 (1.33)	0.366 (1.61)	0.429 (2.04)	0.401 (2.57)	0.289 (3.14)

Table A.15: As Table A.12 but for PESQ with a state-of-the-art DNN based SE algorithm.

	-10dB	-5dB	0dB	5dB	10dB	15dB	20dB
DM bus	0.414 (1.26)	0.550 (1.51)	0.596 (1.93)	0.543 (2.46)	0.401 (3.05)	0.225 (3.61)	0.080 (4.04)
DM cafe	0.007 (1.13)	0.149 (1.11)	0.300 (1.20)	0.425 (1.42)	0.479 (1.83)	0.467 (2.37)	0.372 (2.97)
DM cafeter	-0.047 (1.17)	0.066 (1.11)	0.196 (1.16)	0.350 (1.32)	0.471 (1.65)	0.499 (2.14)	0.423 (2.73)
DM car	**0.811** (1.21)	**1.021** (1.45)	**1.119** (1.81)	**1.005** (2.35)	**0.763** (2.94)	0.447 (3.53)	0.167 (4.01)
DM metro	0.095 (1.13)	0.242 (1.17)	0.373 (1.33)	0.463 (1.65)	0.483 (2.12)	0.408 (2.70)	0.274 (3.28)
DM resto	-0.140 (1.25)	-0.009 (1.13)	0.157 (1.11)	0.338 (1.20)	0.499 (1.43)	0.571 (1.84)	**0.523** (2.4)
DM river	0.111 (1.07)	0.268 (1.09)	0.464 (1.17)	0.639 (1.36)	0.682 (1.75)	**0.615** (2.30)	0.445 (2.92)
DM square	0.170 (1.08)	0.327 (1.14)	0.484 (1.29)	0.572 (1.61)	0.564 (2.10)	0.489 (2.69)	0.343 (3.29)
DM station	0.059 (1.08)	0.199 (1.07)	0.356 (1.13)	0.513 (1.30)	0.610 (1.63)	0.603 (2.14)	0.478 (2.75)
DM traffic	0.172 (1.07)	0.347 (1.09)	0.513 (1.19)	0.620 (1.43)	0.636 (1.86)	0.575 (2.43)	0.427 (3.05)
Average	0.165 (1.14)	0.316 (1.19)	0.456 (1.33)	0.547 (1.61)	0.559 (2.04)	0.490 (2.57)	0.353 (3.14)

improve SI, when exposed to an unseen female speaker. This is an extension of the experiments in [9] where the system was tested in matched speaker and matched SNR conditions only.

Furthermore, DNN-5, which is a "general" system that has been trained on a wide range of speakers, noise types and SNRs, is included in the experiments to investigate if such a general system can improve SI, when exposed to both an unseen speaker and noise type.

The noise types used for training DNN-5 include White Gaussian Noise

Table A.16: DNN based SE systems used for the intelligibility test presented in Figs. A.1 and A.2. The first colum shows the system ID and the remaining columns show the training criteria.

System ID	Noise Dim.	SNR Dim.	Speaker Dim.
DNN-1	SSN	-5 dB	20 Female
DNN-2	SSN	-15 dB – 20 dB	20 Female
DNN-3	BBL	-5 dB	20 Female
DNN-4	BBL	-15 dB – 20 dB	20 Female
DNN-5	N3–N7, WGN, BBL-ADFD	-15 dB – 20 dB	20 Female, 20 Male

(WGN), babble noise (BBL-ADFD) and N3 - N7 from the noise dimension tests described in subsection 3.2. The BBL-ADFD noise is constructed using the procedure for BBL, as described in subsection 3.1, but with three males and three females from the unused part of the ADFD corpus. Each test subject was exposed to five repetitions of 32 test conditions (2 noise types $\times$ 4 SNRs $\times$ 4 processing conditions), hence each test subject was exposed to a total of 160 sentences. The two noise types are SSN (N1) and BBL (N2) noise and the four SNRs are -13 dB, -9 dB, -5 dB and -1 dB. This SNR range was chosen to cover SNRs where SI is close to 0% (-13 dB) and close to 100% (-1 dB). The four processing conditions for each noise type were unprocessed corrupted speech, and corrupted speech processed by DNN-1, DNN-2, and DNN-5, for SSN and DNN-3, DNN-4, and DNN-5, for BBL noise. Immediately prior to the listening test, each test subject performed a familiarization test using 24 noisy utterances from a left out test set. The speech material used for the SI test was based on the Danish Dantale-II speech corpus [69]. Each utterance, which is spoken by a female, consists of five words from five different word classes appearing in the following order: name, verb, numeral, adjective and a noun and the test subject was asked to identify the spoken words via a computer interface. There are a total of 10 different words within each word class, hence the Dantale-II corpus is based on a total of 50 different words. All sentences are constructed such that they are syntactically correct but semantically unlikely, which makes it difficult to predict one word based on another, hence the corpus is suitable for intelligibility tests. The SI test was performed in an audiometric booth using a set of beyerdynamic DT 770 headphones and a Focusrite Scarlett 2i2 sound card

The results are presented in Figs. A.1 and A.2 for SSN and BBL noise, respectively. Figs. A.1 and A.2 show that DNN-5, which is the speaker, noise type, and SNR general system, is unable to improve SI at any of the four SNRs of BBL noise as well as the SNRs at -13 dB, -9 dB, and -1 dB of SSN. A paired-sample t-test shows that this SI degradation is statistical significant, i.e. $p < 0.05$, for all these results. It is also seen that DNN-5 improves SI with a small amount for SSN at an SNR of -5 dB. However, this improvement is not statistically significant ($p = 0.44$). For DNN-2 and DNN-

4, which are the female and noise type specific, but SNR general systems, a somewhat different picture is observed. In general both DNN-2 and DNN-4 perform better than DNN-5. For SSN, DNN-2 manages to improve SI over the unprocessed signals at SNR -9 dB, while DNN-4 improves SI at SNRs of -5 dB and -1 dB. However, none of these improvements are statistical significant ($p = 0.10$, $p = 0.10$, $p = 0.25$, respectively)

Finally, for DNN-1 and DNN-2, which are the female, noise type, and SNR specific systems, DNN-1 improves over DNN-2, whereas DNN-3 in general performs worse than DNN-4. Especially at an SNR of -5 dB DNN-3 performs significantly ($p < 0.001$) worse than DNN-4 ($p = 0.10$) relative to the unprocessed signals. This is surprising since DNN-3 is trained at only -5 dB SNR, while DNN-4 had been trained using the SNR range from -15 dB to 20 dB. Furthermore, the observed SI improvement, especially for DNN-4 and DNN-5 using BBL, is lower than one would expect based on the STOI scores for related models in Sec. 3. This discrepancy between STOI scores and observed SI, especially for highly modulated noise signals, has previously been observed [9, 13, 55, 70]. For DNN-1 a statistically significant improvement of 10.4 percentage points ($p = 0.011$) in SI is observed at an SNR of -5 dB, which also corresponds to the SNR at which DNN-1 is trained. To the authors knowledge, SI improvements achieved by a female specific DNN based SE system tested on an unseen female speaker has not yet been reported. Furthermore, the system outperforms a wide range of previously reported SI test results by non-machine learning based methods reported in [65, 66] and is comparable with the SI results reported in [37] where a single continuous-gain MMSE method was used.

4 Conclusion

In this paper the generalizability of a state-of-the-art Deep Neural Network (DNN) based Speech Enhancement (SE) method has been investigated. Specifically, it has been investigated how noise specific, speaker specific and Signal-to-Noise Ratio (SNR) specific systems perform in relation to noise general, speaker general and SNR general systems, respectively. Furthermore, it has been investigated how such systems perform in relation to a single DNN based SE system which has been designed to be speaker, noise type and SNR general. Also, a comparison between this general DNN based SE system and a state-of-the-art Short-Time Spectral Amplitude Minimum Mean Square Error (STSA-MMSE) based SE method has been conducted. In general, a positive correspondence between training data variability and generalization was observed. Specifically, it was found that DNN based SE systems generalize well to both unseen speakers and unseen noise types given a large number of speakers and noise types were included in the training set. Furthermore,

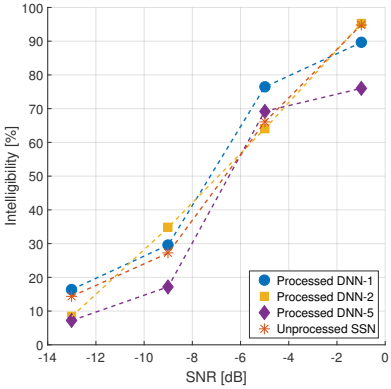


Fig. A.1: SI test results for 3 different DNN based SE systems processing SSN corrupted speech signals based on 10 Danish test subjects.

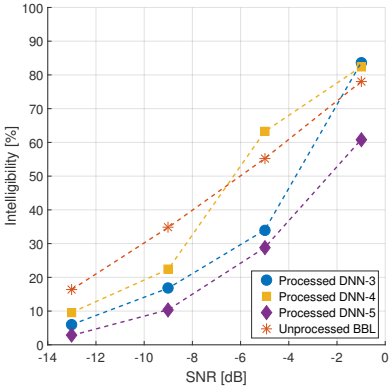


Fig. A.2: SI test results for 3 different DNN based SE systems processing BBL corrupted speech signals based on 10 Danish test subjects.

it was found that specialized DNN based SE systems trained on only one noise type, one speaker or one SNR, outperformed DNN based SE systems trained on a wide range of noise types, speakers, and SNRs in terms of both estimated Speech Quality (SQ) and estimated Speech Intelligibility (SI). In addition, a general DNN based SE algorithm trained using a large number of speakers, a large number of noise types at a large range of SNRs, outperformed a state-of-the-art STSA-MMSE SE algorithm in terms of estimated SQ and SI. However, the performance of this general DNN based SE system, was considerably reduced compared to the specialized systems, that have been optimized to only a single noise type, a single speaker or a single SNR. Finally, it was found that a DNN based SE system trained to be female, noise type and SNR specific, was able to improve SI when tested with an unseen female speaker for particular SNR and noise type configurations, although degrading SI for others.

In general, it can be concluded that DNN based SE systems do have potential to improve SI in a broader range of usage situations than investigated in [9, 13]. Furthermore, the experiments conducted in this paper, indicate that matching the noise type is critical in acquiring good performance for DNN based SE algorithms, whereas matching the SNR dimension is the least critical followed by the speaker dimension for which good generalization can be achieved with a modest amount of training speakers. Also, it can be concluded that considerable improvement in performance can be achieved if the usage situation is limited such that the DNN based SE method can be optimized towards a specific application.

Even though the results reported in this paper are considered general, there is some experimental evidence [13, 16, 20, 46, 53, 54] showing that generalizability performance of DNN based SE algorithms, and DNNs in general, improves when more data and larger networks are being applied, hence SQ and SI performance of DNN based SE systems are expected to improve in the future, when more data and computational resources become available.

Acknowledgment

The authors would like to thank Asger Heidemann Andersen for providing software used to conduct the SI tests, and NVIDIA Corporation for the donation of a Titan X GPU.

References

[1] R. C. Hendriks, T. Gerkmann, and J. Jensen, "DFT-Domain Based Single-Microphone Noise Reduction for Speech Enhancement: A Survey of the State of the Art," *Synthesis Lectures on Speech and Audio Processing*, vol. 9, no. 1, pp. 1–80, Jan. 2013.

[2] P. C. Loizou, *Speech Enhancement: Theory and Practice*. CRC Press, 2013, vol. 2013.

[3] Y. Wang, "Supervised Speech Separation Using Deep Neural Networks," Ph.D. dissertation, The Ohio State University, 2015.

[4] Y. Ephraim and D. Malah, "Speech enhancement using a minimum mean-square error log-spectral amplitude estimator," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 33, no. 2, pp. 443–445, Apr. 1985.

[5] R. Martin, "Speech Enhancement Based on Minimum Mean-Square Error Estimation and Supergaussian Priors," *IEEE Speech Audio Process.*, vol. 13, no. 5, pp. 845–856, Sep. 2005.

[6] J. Erkelens, R. Hendriks, R. Heusdens, and J. Jensen, "Minimum Mean-Square Error Estimation of Discrete Fourier Coefficients With Generalized Gamma Priors," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 15, no. 6, pp. 1741–1752, Aug. 2007.

[7] Y. Wang, A. Narayanan, and D. Wang, "On Training Targets for Supervised Speech Separation," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 22, no. 12, pp. 1849–1858, Dec. 2014.

[8] E. W. Healy, S. E. Yoho, Y. Wang, and D. Wang, "An algorithm to improve speech recognition in noise for hearing-impaired listeners," *J. Acoust. Soc. Am.*, vol. 134, no. 4, pp. 3029–3038, Oct. 2013.

[9] E. W. Healy, S. E. Yoho, J. Chen, Y. Wang, and D. Wang, "An algorithm to increase speech intelligibility for hearing-impaired listeners in novel segments of the same noise type," *J. Acoust. Soc. Am.*, vol. 138, no. 3, pp. 1660–1669, Sep. 2015.

[10] G. Kim, Y. Lu, Y. Hu, and P. C. Loizou, "An algorithm that improves speech intelligibility in noise for normal-hearing listeners," *J. Acoust. Soc. Am.*, vol. 126, no. 3, pp. 1486–1494, Sep. 2009.

[11] K. Han and D. Wang, "A classification based approach to speech segregation," *J. Acoust. Soc. Am.*, vol. 132, no. 5, pp. 3475–3483, Nov. 2012.

[12] Y. Wang and D. Wang, "Towards Scaling Up Classification-Based Speech Separation," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 21, no. 7, pp. 1381–1390, Jul. 2013.

[13] J. Chen, Y. Wang, S. E. Yoho, D. Wang, and E. W. Healy, "Large-scale training to increase speech intelligibility for hearing-impaired listeners in novel noises," *The Journal of the Acoustical Society of America*, vol. 139, no. 5, pp. 2604–2612, May 2016.

[14] J. Chen, Y. Wang, and D. Wang, "Noise perturbation for supervised speech separation," *Speech Communication*, vol. 78, pp. 1–10, 2016.

[15] K. Han and D. Wang, "Towards Generalizing Classification Based Speech Separation," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 21, no. 1, pp. 168–177, Jan. 2013.

[16] Y. Xu, J. Du, L.-R. Dai, and C.-H. Lee, "A Regression Approach to Speech Enhancement Based on Deep Neural Networks," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 23, no. 1, pp. 7–19, Jan. 2015.

[17] T. Lee and F. Theunissen, "A single microphone noise reduction algorithm based on the detection and reconstruction of spectro-temporal features," *Proc. R. Soc. A*, vol. 471, no. 2184, Dec. 2015.

[18] P.-S. Huang, M. Kim, M. Hasegawa-Johnson, and P. Smaragdis, "Joint Optimization of Masks and Deep Recurrent Neural Networks for Monaural Source Separation," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 23, no. 12, pp. 2136–2147, Dec. 2015.

[19] D. Liu, P. Smaragdis, and M. Kim, "Experiments on Deep Learning for Speech Denoising," in *INTERSPEECH*, 2014.

[20] Y. Wang, J. Chen, and D. Wang, "Deep neural network based supervised speech segregation generalizes to novel noises through large-scale training," Department of Computer Science and Engineering, The Ohio State University, Tech. Rep. OSU-CISRC-3/15-TR02, 2015.

[21] S. Gonzalez and M. Brookes, "Mask-based enhancement for very low quality speech," in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2014, pp. 7029–7033.

[22] J. Chen and D. Wang, "Long Short-Term Memory for Speaker Generalization in Supervised Speech Separation," in *INTERSPEECH 2016, 17th Annual Conference of the International Speech Communication Association*, 2016, pp. 3314 – 3318.

[23] M. Delfarah and D. Wang, "A feature study for masking-based reverberant speech separation," in *INTERSPEECH 2016, 17th Annual Conference of the International Speech Communication Association*, 2016, pp. 555 – 559.

[24] A. Kumar and D. Florencio, "Speech Enhancement In Multiple-Noise Conditions using Deep Neural Networks," *arXiv:1605.02427 [cs]*, May 2016, arXiv: 1605.02427.

[25] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer, 2006, vol. 2006.

[26] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, May 2015.

[27] J. S. Garofolo, L. F. Lamel, W. M. Fisher, J. G. Fiscus, D. S. Pallett, and N. L. Dahlgren, "DARPA TIMIT Acoustic Phonetic Continuous Speech Corpus CDROM," 1993.

[28] T. May and T. Dau, "Requirements for the evaluation of computational speech segregation systems," *J. Acoust. Soc. Am.*, vol. 136, no. 6, pp. EL398–EL404, Dec. 2014.

[29] Y. Wang and D. Wang, "A deep neural network for time-domain signal reconstruction," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, Apr. 2015, pp. 4390–4394.

[30] I. Cohen, "Noise spectrum estimation in adverse environments: improved minima controlled recursive averaging," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 11, no. 5, pp. 466–475, Sep. 2003.

[31] I. Cohen and B. Berdugo, "Speech enhancement for non-stationary noise environments," *Signal Processing*, vol. 81, no. 11, pp. 2403–2418, Nov. 2001.

[32] J. Erkelens, R. Hendriks, and R. Heusdens, "On the Estimation of Complex Speech DFT Coefficients Without Assuming Independent Real and Imaginary Parts," *IEEE Signal Processing Letters*, vol. 15, pp. 213–216, 2008.

[33] R. Hendriks, J. Erkelens, and R. Heusdens, "Comparison of complex-DFT estimators with and without the independence assumption of real and imaginary parts," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, Mar. 2008, pp. 4033–4036.

[34] D. D. Lee and H. S. Seung, "Algorithms for Non-negative Matrix Factorization," *Neural Information Processing Systems - NIPS*, vol. 13, pp. 556–562, 2000.

[35] Y. Xu, J. Du, L.-R. Dai, and C.-H. Lee, "An Experimental Study on Speech Enhancement Based on Deep Neural Networks," *IEEE Signal Processing Letters*, vol. 21, no. 1, pp. 65–68, Jan. 2014.

[36] D. Wang, "Time-frequency masking for speech separation and its potential for hearing aid design," *Trends Amplif.*, vol. 12, no. 4, pp. 332–353, Dec. 2008.

[37] J. Jensen and R. Hendriks, "Spectral Magnitude Minimum Mean-Square Error Estimation Using Binary and Continuous Gain Functions," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 20, no. 1, pp. 92–102, Jan. 2012.

[38] C. Hummersone, T. Stokes, and T. Brookes, "On the Ideal Ratio Mask as the Goal of Computational Auditory Scene Analysis," in *Blind Source Separation*, ser. Signals and Communication Technology, G. R. Naik and W. Wang, Eds. Springer Berlin Heidelberg, 2014, pp. 349–368.

[39] D. Wang, "On Ideal Binary Mask As the Computational Goal of Auditory Scene Analysis," in *Speech Separation by Humans and Machines*, P. Divenyi, Ed. Springer US, 2005, pp. 181–197.

[40] N. Madhu, A. Spriet, S. Jansen, R. Koning, and J. Wouters, "The Potential for Speech Intelligibility Improvement Using the Ideal Binary Mask and the Ideal Wiener Filter in Single Channel Noise Reduction Systems: Application to Auditory Prostheses," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 21, no. 1, pp. 63–72, Jan. 2013.

[41] D. Williamson, Y. Wang, and D. Wang, "Complex Ratio Masking for Monaural Speech Separation," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. PP, no. 99, pp. 1–1, 2015.

[42] D. Wang and G. J. Brown, *Computational Auditory Scene Analysis: Principles, Algorithms, and Applications*. Wiley-IEEE Press, 2006.

[43] E. W. Healy, S. E. Yoho, Y. Wang, F. Apoux, and D. Wang, "Speech-cue transmission by an algorithm to increase consonant recognition in noise for hearing-impaired listeners," *J. Acoust. Soc. Am.*, vol. 136, no. 6, pp. 3325–3336, Dec. 2014.

[44] J. Chen, Y. Wang, and D. Wang, "A Feature Study for Classification-Based Speech Separation at Low Signal-to-Noise Ratios," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 22, no. 12, pp. 1993–2002, Dec. 2014.

[45] Y. Wang, K. Han, and D. Wang, "Exploring Monaural Features for Classification-Based Speech Segregation," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 21, no. 2, pp. 270–279, Feb. 2013.

[46] D. Amodei *et al.*, "Deep Speech 2: End-to-End Speech Recognition in English and Mandarin," *arXiv:1512.02595 [cs]*, Dec. 2015.

[47] V. Nair and G. E. Hinton, "Rectified Linear Units Improve Restricted Boltzmann Machines," in *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, 2010, pp. 807–814.

[48] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proc. Int. Conf. Artificial Intelligence and Statistics*, 2010.

[49] J. Duchi, E. Hazan, and Y. Singer, "Adaptive Subgradient Methods for Online Learning and Stochastic Optimization," *J. Mach. Learn. Res.*, vol. 12, pp. 2121–2159, Jul. 2011.

[50] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Improving neural networks by preventing co-adaptation of feature detectors," *arXiv: 1207.0580*, Jul. 2012.

[51] A. Mohamed, G. E. Dahl, and G. Hinton, "Acoustic Modeling Using Deep Belief Networks," *Trans. Audio, Speech and Lang. Proc.*, vol. 20, no. 1, pp. 14–22, Jan. 2012.

[52] Y. Bengio, P. Lamblin, D. Popovici, H. Larochelle, U. D. Montréal, and M. Québec, "Greedy Layer-Wise Training of Deep Networks," in *Advances in Neural Information Processing Systems 19*. MIT Press, 2007, pp. 153–160.

[53] A. Halevy, P. Norvig, and F. Pereira, "The Unreasonable Effectiveness of Data," *IEEE Intelligent Systems*, vol. 24, no. 2, pp. 8–12, Mar. 2009.

[54] A. Hannun *et al.*, "Deep Speech: Scaling up end-to-end speech recognition," *arXiv:1412.5567 [cs]*, Dec. 2014.

[55] C. Taal, R. Hendriks, R. Heusdens, and J. Jensen, "An Algorithm for Intelligibility Prediction of Time-Frequency Weighted Noisy Speech," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 19, no. 7, pp. 2125–2136, Sep. 2011.

[56] A. Rix, J. Beerends, M. Hollier, and A. Hekstra, "Perceptual evaluation of speech quality (PESQ)-a new method for speech quality assessment of telephone networks and codecs," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, vol. 2, 2001, pp. 749–752 vol.2.

[57] "P.862.2 : Wideband extension to Recommendation P.862 for the assessment of wideband telephone networks and speech codecs," 2005. [Online]. Available: https://www.itu.int/rec/T-REC-P.862.2-200511-S/en

[58] C. Févotte, R. Gribonval, and E. Vincent, "BSS EVAL Toolbox User Guide – Revision 2.0," IRISA, Tech. Rep. inria-00564760, 2011. [Online]. Available: https://hal.inria.fr/inria-00564760

[59] X. L. Zhang and D. Wang, "A Deep Ensemble Learning Method for Monaural Speech Separation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 5, pp. 967–977, May 2016.

[60] J. Barker, M. Ricard, V. Vincent, and S. Watanabe, "The third 'CHiME' Speech Separation and Recognition Challenge: Dataset, task and baselines," *ASRU*, 2015.

[61] D. Williamson, Y. Wang, and D. Wang, "Deep neural networks for estimating speech model activations," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, Apr. 2015, pp. 5113–5117.

[62] M. Kolbæk, Z.-H. Tan, and J. Jensen, "Speech enhancement using long short-term memory based recurrent neural networks for noise robust speaker verification," in *Spoken Language Technology Workshop (SLT), 2016 IEEE*, 2016.

[63] D. S. Williamson, Y. Wang, and D. Wang, "Estimating nonnegative matrix model activations with deep neural networks to increase perceptual speech quality," *J. Acoust. Soc. Am.*, vol. 138, no. 3, pp. 1399–1407, Sep. 2015.

[64] Y. Ephraim and D. Malah, "Speech enhancement using a minimum-mean square error short-time spectral amplitude estimator," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 32, no. 6, pp. 1109–1121, Dec. 1984.

[65] Y. Hu and P. C. Loizou, "A comparative intelligibility study of single-microphone noise reduction algorithms," *J. Acoust. Soc. Am.*, vol. 122, no. 3, pp. 1777–1786, Sep. 2007.

[66] H. Luts *et al.*, "Multicenter evaluation of signal enhancement algorithms for hearing aids," *J. Acoust. Soc. Am.*, vol. 127, no. 3, pp. 1491–1505, Mar. 2010.

[67] R. Hendriks, R. Heusdens, and J. Jensen, "MMSE based noise PSD tracking with low complexity," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, Mar. 2010, pp. 4266–4269.

[68] M. Seltzer, D. Yu, and Y. Wang, "An investigation of deep neural networks for noise robust speech recognition," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, May 2013, pp. 7398–7402.

[69] K. Wagener, J. L. Josvassen, and R. Ardenkjaer, "Design, optimization and evaluation of a Danish sentence test in noise," *Int. J. Audiol.*, vol. 42, no. 1, pp. 10–17, Jan. 2003.

[70] S. Jørgensen, R. Decorsière, and T. Dau, "Effects of manipulating the signal-to-noise envelope power ratio on speech intelligibility," *The Journal of the Acoustical Society of America*, vol. 137, no. 3, pp. 1401–1410, Mar. 2015.

Paper B

Speech Enhancement Using Long Short-Term Memory Based Recurrent Neural Networks for Noise Robust Speaker Verification

Morten Kolbæk, Zheng-Hua Tan, and Jesper Jensen

The paper has been published in
Proceedings IEEE Spoken Language Technology Workshop,
pp. 305-311, December 2016.

© 2016 IEEE
The layout has been revised.

Abstract

In this paper we propose to use a state-of-the-art Deep Recurrent Neural Network (DRNN) based Speech Enhancement (SE) algorithm for noise robust Speaker Verification (SV). Specifically, we study the performance of an i-vector based SV system, when tested in noisy conditions using a DRNN based SE front-end utilizing a Long Short-Term Memory (LSTM) architecture. We make comparisons to systems using a Non-negative Matrix Factorization (NMF) based front-end, and a Short-Time Spectral Amplitude Minimum Mean Square Error (STSA-MMSE) based front-end, respectively.

We show in simulation experiments that a male-speaker and text-independent DRNN based SE front-end, without specific a priori *knowledge about the noise type outperforms a text, noise type and speaker dependent NMF based front-end as well as a STSA-MMSE based front-end in terms of equal error rates for a large range of noise types and signal to noise ratios on the RSR2015 speech corpus.*

1 Introduction

Biometric technologies, such as speaker verification (SV), are a secure, fast and convenient alternative to traditional authentication methods such as typed passwords. In fact, the global market for biometric technologies is rapidly growing and is expected to reach $41.5 billion in 2020 with annual growth rates of more than 20% [1]. However, before biometric technologies can be completely adopted and applied in practice, they must, among other things, be robust against external interferences. This implies that the SV systems are reliable in a broad range of acoustic settings including different noisy environments, competing talkers, recording devices, etc.

In recent years, the branch of machine learning known as Deep Learning (DL) has gained a tremendous amount of attention in both academia and industry. DL is a term covering a wide range of machine learning techniques such as Deep Neural Networks (DNN), Recurrent Neural Networks or Convolutional Neural Networks (CNN) [2]. Techniques, which have revolutionized a wide range of applications. [3–7].

Especially Automatic Speech Recognition (ASR) has been improved using DL [8, 9] and, although DL has revolutionized ASR, DL based noise robust SV has not attained much attention [10–13].

In general, SV is the task of verifying the identity of a person based on the voice of the speaker. Specifically, a SV system records via a single or multiple microphones an utterance from a speaker, and the task of the SV system is to verify or reject the claimed identity based on this spoken utterance. If the spoken text is known *a priori*, it is referred as text-dependent SV, while if the spoken text is unknown it is referred as text-independent SV.

In this paper, noise robustness of a text-dependent SV system is investigated using a state-of-the-art Long Short-Term Memory (LSTM) based Deep Recurrent Neural Network (DRNN) applied as a denoising front-end using single microphone recordings. It should be mentioned that Speaker Recognition (SR) and SV is very related and differ in principle only in the way the system is evaluated or applied, i.e. for either identity verification or recognition. Although, we focus on SV in this paper the proposed front-end denoising techniques could just as well be applied for SR. For the same reasons the referenced literature focuses on both SR and SV.

Typically, noise robust SV systems can be achieved by modifying either the back-end or the front-end of the SV system [10–30]. The back-end constitutes the Universal Background Model (UBM), i-vector extractor and scoring, whereas the front-end constitutes preprocessing steps in terms of denoising of the microphone signal, and feature extraction, prior to back-end processing. Even though noise robust SV has been intensively studied in the literature [10–25, 27–30], only a few studies [10–13] have applied DNNs in a denoising context. Furthermore, none of these studies apply a DRNN as a SE front-end and compare these methods with existing SE approaches.

In a recent study [20] it was shown that if *a priori* knowledge about the noise type is available, a Non-negative Matrix Factorization (NMF) based SE front-end outperforms a Wiener filtering based SE front-end [31] as well as a Short-Time Spectral Amplitude Minimum Mean Square Error (STSA-MMSE) based SE front-end [32]. However, in the SE literature, several studies [33–35] show that DNN based SE algorithms outperform NMF based SE methods in terms of estimated speech quality and estimated speech intelligibility. Hence, a natural question to ask is whether DNN based SE algorithms also outperform NMF based SE in a SV context. This is the question addressed in this paper.

The paper is organized as follows. In Sec. 2 the speech corpora and noise data used for training and testing the NMF dictionaries, the DRNN models as well as the SV system are described. In Sec. 3 the proposed DRNN based SE front-ends are presented and in Sec. 4 the baseline systems are presented, which are the NMF based SE front-ends, the STSA-MMSE based SE front-end as well as the SV baseline system. In Sec. 5 the experimental design and results are discussed and finally the paper is concluded in Sec. 6.

2 Speech and Noise Data

The denoising task performed by the SE front-ends investigated in this paper can be described by the linear model given by

$$y(n) = x(n) + d(n), \tag{B.1}$$

where $y(n)$, $x(n)$, and $d(n)$ are the noisy speech signal, the clean speech signal and the additive noise signal, respectively. The task of the denoising front-ends, further described in the following sections, is to estimate $x(n)$ based on observations of $y(n)$.

2.1 Speech Corpora

In all simulation experiments (reported in Sec. 5) the clean speech signal $x(n)$, is based on the male part of the RSR2015 corpus [36] and the data is allocated among the SV speaker models, the NMF, and DRNN front-ends according to Table. B.1.

For training SV speaker models, text ID 1 and sessions 1, 4, and 7 from male speakers from m002 to m050 and m052 are selected, and for testing, sessions 2, 3, 5, 6, 8, and 9 are used. Hence, the SV system is text-dependent and is based on 50 male speakers and each speaker is enrolled in the system using 3 utterances and tested with 6 utterances. Furthermore, sessions 1, 4, and 7 have been recorded using a Samsung Nexus smartphone, whereas sessions 2, 5, and 8 and 3, 6, and 9 have been recorded using a Samsung Galaxy S and a HTC Desire smartphone, respectively. That is, the SV system is tested in an unmatched microphone/recording device setting.

For training the speaker dependent dictionaries used by the NMF based SE front-ends, text ID 1 and sessions 1, 4, and 7 are used. Hence, the NMF font-ends are similarly tested in an unmatched microphone/recording setting.

The DRNN based front-ends are trained using text IDs 2 – 30 and sessions 1 – 9 from male speakers from m053 – m142, and validated in terms of an early stopping scheme using the same utterances and session IDs, but using speakers from m143 – m157. Although the DRNN front-ends are tested in a matched microphone setting, since they are trained on all nine sessions, they are tested in an unmatched text and an unmatched speaker setting, which is considered a considerably more challenging task.

Table B.1: Allocation of RSR2015 male-speaker speech data used for training and testing the SV system, as well as the NMF and DRNN front-ends.

System	Cond.	Text ID.	Sess. ID	Sprk. ID
SV	Train	1	1, 4, 7	2 – 50 & 52
SV	Test	1	2,3,5,6,8,9	2 – 50 & 52
NMF	Train	1	1, 4, 7	2 – 50 & 52
NMF	Test	1	2,3,5,6,8,9	2 – 50 & 52
DRNN	Train	2–30	1–9	53 – 142
DRNN	Val	2–30	1–9	143 – 157
DRNN	Test	1	2,3,5,6,8,9	2 – 50 & 52

2.2 Noise Data

The noise signal $d(n)$ (as given by Eq. (B.1)) is used to simulate real-life noisy environments, such that the noise robustness of the SV system can be evaluated. For this evaluation the following 6 noise types are used: bus (BUS), cafeteria (CAF), street (STR) and pedestrian (PED) from the CHiME3 dataset [37], as well as a babble (BBL) noise, and a Speech Shaped Noise (SSN) created by the authors.

The SSN sequence is constructed by filtering a 50 min. Gaussian white noise sequence through a 12th-order all-pole filter with coefficients found from Linear Predictive Coding (LPC) analysis of 100 randomly chosen sentences from a Danish speech corpus known as *Akustiske Databaser for Dansk* (ADFD)[1].

The BBL noise is similarly based on the ADFD corpus. From the ADFD test set, four male and four female speakers are randomly selected. Each speaker is represented by 986 utterances which are normalized to unit Root Mean Square (RMS) following the removal of any silent segments using a Voice Activity Detection (VAD) algorithm. Then, all 986 utterances from each speaker is concatenated into 8 signals, following truncation to equal length and addition of the eight signals into a single eight speaker babble noise signal.

All six (BUS, CAF, STR, PED, BBL, SSN) noise types were first truncated to have a total duration of 50 min. and then divided into a 40 min. training set, 5 min. validation set and a 5 min. test set. Hence, there are no overlapping noise segments between the training, validation, and test noise.

The noisy mixtures at different Signal to Noise Ratios (SNRs) were constructed using the model in Eq. (B.1) by scaling the noise signal $d(n)$ accordingly. The noise signal was scaled to achieve the desired SNR based on the duration of the entire speech signal $x(n)$.

Furthermore, a sampling frequency of 16 kHz is used throughout the paper and all audio files are normalized to unit RMS.

3 Speech Enhancement Using Deep Recurrent Neural Networks

Speech enhancement algorithms based on DNNs have in recent years gained a large amount of attention and showed impressive performance in terms of improving speech quality and speech intelligibility [33–35, 38, 39]. Common for these algorithms is that they use a DNN as a regression model to estimate a ratio mask that is applied to the Time-Frequency (T-F) representation of the noisy speech signal to acquire an estimate of the clean speech signal.

[1] https://www.nb.no/sbfil/dok/nst_taledat_no.pdf

A related approach will be adopted in this paper. Specifically, a DNN is employed, but it is improved using LSTM layers [40] and a training criterion that indirectly constructs a ratio mask by minimizing the Mean Square Error (MSE) between the desired clean speech signal and the noise. In this way, the model learns to separate speech from noise, which is the real desired goal, rather than minimizing the MSE between an ideal mask and an estimated mask, as is typically done [38, 41, 42].

The T-F representation used for the DRNN based SE front-end is a $N_{STFT} = 512$ point Short-Time Fourier Transform (STFT) using a frame width of 32 ms and a frame spacing of 16 ms [31]. In this way, a frequency dimension of $N = N_{STFT}/2 + 1 = 257$, covering positive frequencies, is achieved. When the estimated ratio mask has been applied to the noisy speech signal, the time domain representation is achieved by applying an Inverse Short-Time Fourier Transform (ISTFT) using the phase from the noisy signal.

3.1 DRNN Architecture and Training

All DRNN based front-ends are based on an architecture constituting two LSTM layers and a single fully connected feed-forward output layer with sigmoid activation functions. The input to the DRNN is the magnitude of the STFT coefficients of the noisy mixture $y(n)$, including a context of 15 past frames and 15 future frames, hence arriving at a final input dimension of $N \times 31 = 257 \times 31 = 7967$. The output constitutes a ratio mask for a single frame, and the dimension is therefore related to the size of the STFT, i.e. 257 (STFT order is 512).

The training criterion used for training the DRNNs is defined as follows: Let $|x(n,\omega)|$, $|d(n,\omega)|$ and $|y(n,\omega)|$ denote the magnitude of the STFT of the clean speech signal, the noise signal and the noisy mixture, respectively. Furthermore let $\hat{x}(n,\omega)$ and $\hat{d}(n,\omega)$ denote the estimate of the magnitude of the clean speech signal and noise signal, respectively. Finally, let $o(n,\omega)$ denote the output of the DRNN, and let $m_x(n,\omega)$ and $m_d(n,\omega)$ denote the ratio mask representing the speech signal and noise signal, respectively. Since the DRNN has one sigmoid output layer, the speech ratio mask $m_x(n,\omega)$ for a single T-F unit is simply defined as

$$m_x(n,\omega) = o(n,\omega), \tag{B.2}$$

and $m_d(n,\omega)$ as

$$m_d(n,\omega) = 1 - o(n,\omega). \tag{B.3}$$

Furthermore, $\hat{x}(n,\omega)$ is defined as

$$\hat{x}(n,\omega) = m_x(n,\omega) \times |y(n,\omega)|, \tag{B.4}$$

and $\hat{d}(n,\omega)$ as

$$\hat{d}(n,\omega) = m_d(n,\omega) \times |y(n,\omega)|. \tag{B.5}$$

Finally, the DRNN MSE training criteria for a single training example ($d(n,\omega)$, $x(n,\omega)$) is defined as:

$$\begin{aligned} MSE(n) = &\frac{1}{N}\sum_{\omega=1}^{N}(\hat{d}(n,\omega) - |d(n,\omega)|)^2 \\ &+ \frac{1}{N}\sum_{\omega=1}^{N}(\hat{x}(n,\omega) - |x(n,\omega)|)^2. \end{aligned} \tag{B.6}$$

By using the training criteria given by Eq. (B.6), it is ensured that the MSE between $\hat{d}(n,\omega)$ and $|d(n,\omega)|$, as well as $\hat{x}(n,\omega)$ and $|x(n,\omega)|$ is minimized, while still ensuring that $m_d(n,\omega)$ and $m_x(n,\omega)$ represents a valid ratio mask, i.e. $m_x(n,\omega) + m_d(n,\omega) = 1$.

Although $m_x(n,\omega)$ and $m_d(n,\omega)$ are not explicitly used, in this particular work, since the current DRNN directly estimates $|x(n,\omega)|$, the formulation in Eq. (B.6) allows the output layer to be straightforwardly extended to multiple outputs by extending the dimension of the output layer and applying a softmax to ensure all outputs are correctly normalized, hence separating e.g. multiple speakers [43].

The DRNNs used for all experiments in this paper are implemented in CNTK[2] [44] and are trained using stochastic gradient descent with truncated backpropagation through time, using 10 time steps and a momentum term of 0.9 for all epochs. The learning rate is initially set to 0.1, but is reduced with a factor of 2, when the validation error has not decreased for one epoch. During training, 20% dropout [45] is used for the LSTM layers and the training is aborted, when the learning rate becomes less than 1^{-10}. When the learning rate is decreased, the training continues from the previous best model.

3.2 DRNN Based SE Front-Ends

A total of seven DRNN based SE front-ends are investigated: Six Noise Specific DRNN (NSDRNN) front-ends, one for each noise type, and one Noise General DRNN (NGDRNN) front-end trained on all six noise types.

The NSDRNN front-ends are each trained on a particular noise type, hence, at test time, *a priori* knowledge about the noise type is required. This is similar to the NMF front-ends, which also rely on this prior knowledge.

For the NGDRNN front-end, only a single model is trained using a combination of all six noise types. This front-end therefore utilizes only a minimum amount of *a priori* information, since it is unaware of the actual noise type. The NGDRNN front-end is included to investigate the performance that can be achieved, if less *a priori* knowledge about the noise type is available.

[2] https://www.cntk.ai

For all DRNN front-ends, 10^5 noisy mixtures are used for training. The mixtures are generated by drawing a SNR at random from a discrete uniform distribution defined within the SNR range from -5 dB – 20 dB. Due to the large number of realizations, it is assumed that the distribution of drawn SNRs is approximately uniform. The noise signal used for each noisy mixture was extracted from the whole training noise sequence by using a starting index drawn from a discrete uniform distribution defined over the entire length of the noise sequence. If the starting index is such that there is no room for the whole utterance, the remaining samples are extracted from the beginning of the noise sequence. For the NGDRNN front-end, the training noise sequence is constructed by concatenating the six individual noise type sequences, hence the 10^5 noisy mixtures contain all six noise types evenly distributed, whereas for the NSDRNN front-ends the 10^5 noisy mixtures contain only a single noise type. A similar approach is used for generating the mixtures used for validation and test.

4 Baseline Systems

This section describes the SV baseline as well as the SE baseline front-ends, namely the NMF based SE front-ends and the STSA-MMSE based SE front-end.

4.1 NMF Baseline

The basic observation behind NMF is that a non-negative matrix $\mathbf{V} \in \mathbb{R}^{m \times n}$ can be approximately factorized into a product of two non-negative matrices $\mathbf{D} \in \mathbb{R}^{m \times k}$ and $\mathbf{H} \in \mathbb{R}^{k \times n}$ [46] as given by

$$\mathbf{V} \approx \mathbf{DH}, \tag{B.7}$$

where $\mathbf{D}$ is known as the dictionary and $\mathbf{H}$ is the activation matrix. The activation matrix $\mathbf{H}$ is used to identify what parts of the dictionary are required to accurately approximate $\mathbf{V}$. The number of columns k in the dictionary $\mathbf{D}$ is a tuning parameter used to adjust the representational power of the factorization.

The dictionary $\mathbf{D}$ and the activations $\mathbf{H}$ can be found by solving the constrained and regularized least squares minimization problem given by

$$\begin{aligned} \underset{\mathbf{D},\mathbf{H}}{\text{minimize}} \quad & \frac{1}{2}\|\mathbf{V} - \mathbf{DH}\|_F^2 + \alpha\|\mathbf{H}\|_1 \\ \text{subject to} \quad & \mathbf{D}, \mathbf{H} \geq 0, \end{aligned} \tag{B.8}$$

where $\|\cdot\|_F^2$ is the squared Frobenius norm, $\|\cdot\|_1$ is the $\ell 1$-norm, and $\alpha > 0$ is a sparsity parameter [47]. Equation (B.8) can be solved in an iterative fashion

using

$$\mathbf{H} = \mathbf{H} \circ \frac{\mathbf{D}^T\mathbf{V}}{\mathbf{D}^T\mathbf{D}\mathbf{H} + \alpha}, \tag{B.9a}$$

$$\mathbf{D} = \mathbf{D} \circ \frac{\mathbf{V}\mathbf{H}^T}{\mathbf{D}\mathbf{H}\mathbf{H}^T}, \tag{B.9b}$$

where $\circ$ is the Hadamard product, i.e. element-wise multiplication. The solution to Eq. (B.8) is found by alternating between update rule (B.9a) and (B.9b) until the value of the cost function given by Eq. (B.8) is below a predefined threshold [46, 47].

When NMF is applied for SE using a model given by Eq. (B.1), $\mathbf{V}$ is the STFT magnitudes of a noisy speech signal $\mathbf{V_y}$, and is on the following form:

$$\mathbf{V_y} \approx \mathbf{D}\mathbf{H} = \begin{bmatrix}\mathbf{D_x} & \mathbf{D_d}\end{bmatrix} \begin{bmatrix}\mathbf{H_x} \\ \mathbf{H_d}\end{bmatrix}, \tag{B.10}$$

where $\mathbf{D_x}$ and $\mathbf{D_d}$ are speech and noise dictionaries, respectively and $\mathbf{H_x}$ and $\mathbf{H_d}$ are their corresponding activations. The dictionaries $\mathbf{D_x}$ and $\mathbf{D_d}$ are found using the approach given by Eq. (B.9), in an offline training procedure, prior to test time.

At test time, using the already trained $\mathbf{D_x}$ and $\mathbf{D_d}$ and a test sample $\mathbf{V_y}$, the corresponding activations $\mathbf{H_x}$ and $\mathbf{H_d}$ are found jointly using Eqs. (B.10) and (B.9a), and the estimate of the clean speech STFT magnitudes $\hat{\mathbf{X}}$ are acquired by [20]

$$\hat{\mathbf{X}} = \mathbf{Y} \circ \frac{\frac{\mathbf{D_x}\mathbf{H_x}}{\mathbf{D_d}\mathbf{H_d}}}{1 + \frac{\mathbf{D_x}\mathbf{H_x}}{\mathbf{D_d}\mathbf{H_d}}}. \tag{B.11}$$

The time domain signal is finally achieved by ISTFT of $\hat{\mathbf{X}}$ using the phase of the noisy mixture.

For the experiments conducted in this paper, one NMF dictionary $\mathbf{D_x}$, is trained for each speaker and each noise type, hence the NMF front-ends are speaker, text, and noise type dependent. This is similar to the study in [20], which enables the use of a NMF based denoising front-end with only a small amount of training data. However, it requires *a priori* knowledge about the noise type at test time.

Furthermore, similarly to [20], the speaker dictionaries $\mathbf{D_x}$ have a fixed size of 64 columns, i.e. $k = 64$ and are trained using speech-only regions by removing all frames with a sample variance less than 3×10^{-5}. Finally, the NMF training is terminated when the value of the cost function given in Eq. (B.8) is less than 10^{-4} or the number of iterations exceed 500.

4.2 STSA-MMSE Baseline

The STSA-MMSE front-end is a statistical based SE method, which relies on the assumption that noise free Discrete Fourier Transform (DFT) coefficients

are distributed according to a generalized gamma distribution with parameters $\gamma = 2$ and $\nu = 0.15$ [32, 48]. The *a priori* SNR estimator used by the STSA-MMSE front-end is the Decision-Directed approach [49] using a smoothing factor of 0.98 and a noise Power Spectral Density (PSD) estimate based on the noise PSD tracker reported in [50][3]. For each utterance, the noise tracker was initialized using a noise PSD estimate based on the first 1000 samples i.e. 62.5 ms, which is assumed to be a noise-only region. Since the STSA-MMSE front-end only relies on simple statistical assumptions, it is basically text, speaker, and noise type independent and is therefore the method that relies on the least amount of *a priori* knowledge compared to the NMF, NSDRNN, and NGDRNN front-ends.

4.3 Speaker Verification Baseline

The SV baseline is a Gaussian Mixture Model (GMM)-UBM i-vector based system [51] and is similar to the system investigated in [20]. The baseline is implemented using the Kaldi Speech Recognition Toolkit [52].

The 4380 male speaker utterances from the TIMIT corpus [53] are used for obtaining the GMM-UBM as well as the total variability matrix used for i-vector extraction. The used features are 13 Mel Frequency Cepstrum Coefficients (MFCC) based on voice only regions of frames with a duration of 25 ms and a frequency range from 0 – 8 kHz and with cepstral liftering disabled. The energy threshold and the energy mean scale of the Kaldi VAD function are set to their default values of 5.5 and 0.5, respectively.

During enrollment, an i-vector of dimension 400 is generated for each of the three enrollment utterances, for each speaker. The final speaker model is constructed as the average of these three i-vectors.

During test time, the cosine distance between each speaker model i-vector and all test utterance i-vectors is computed, and since 50 speakers are enrolled in the SV system and each speaker is represented by 6 test utterances, a total number of $50 \times 6 \times 50 = 15000$ trials are conducted for each evaluation. When a model is chosen as the target speaker, the remaining 49 models are used as imposters, hence the last multiplication with 50.

From the 15000 cosine scores the Receiver Operating Characteristics (ROC) curve is constructed, and the Equal Error Rate (EER) is identified and used as the final evaluation score. The EER is the location on the ROC curve where the false positive rate is equal to the false rejection rate, i.e. one minus the true positive rate. For EERs, lower is better, and a flawless SV system will achieve an EER of zero.

[3] http://insy.ewi.tudelft.nl/content/software-and-data

5 Experimental Results and Discussion

The performance of the SV system with different denoising front-ends is presented in Tables B.2 – B.7. The SV system is evaluated using noisy mixtures contaminated with the six noise types described in Sec. 2 at SNRs in the range from -5 dB – 20 dB. The system is also evaluated using the clean speech signals without any noise, in order to investigate how the denoising front-ends operate in noise-free conditions.

For each noise type and SNR, the SV system is evaluated using the following five front-ends: No front-end processing (No Proc.), STSA-MMSE based front-end processing, NMF based front-end processing, NSDRNN based front-end processing, and finally NGDRNN based front-end processing.

It should be mentioned that both the NSDRNN and NGDRNN front-ends are tested in unmatched text and speaker conditions, while the NMF method is tested in matched text and speaker conditions.

Furthermore, since the STSA-MMSE, NSDRNN and NGDRNN front-ends are speaker independent the same front-end can be used for both target speakers and imposters. However, since the NMF based front-ends are speaker dependent the NMF front-end with speaker ID similar to the target speaker is used to process all trials for that particular speaker, i.e. for both target speaker and imposters. This is done to account for the situation where the claimed speaker ID is false, i.e. an imposter. In these situations it should be ensured that the NMF processing cannot induce a false positive by using a front-end not matched to the speaker ID.

Table B.2: EER for SV system using BBL noise corrupted speech.

SNR	No Proc.	STSA-MMSE	NMF	NS DRNN	NG DRNN
-5 dB	46.0	44.8	40.1	**28.9**	33.6
0 dB	37.9	36.6	32.2	**19.6**	21.0
5 dB	26.6	27.4	23.8	**14.6**	14.8
10 dB	17.6	18.3	17.0	**12.0**	13.0
15 dB	11.6	12.1	14.0	10.7	**10.5**
20 dB	**9.26**	10.3	11.6	9.39	9.67
Clean	**6.67**	11.7	14.5	10.7	11.7
Average	22.2	23.0	21.9	**15.1**	16.3

It is seen from Tables B.2 – B.7 that the NSDRNN and NGDRNN front-ends achieve the lowest EER for the majority of the test conditions and outperforms the NMF and STSA-MMSE front-ends with a large margin, especially at SNRs below 10 dB. However, no front-end achieves the EER of 6.67 for the clean condition, hence it seems that all methods introduce some distortion at high SNRs. For practical applications it might be beneficial to in-

Table B.3: EER for SV system using BUS noise corrupted speech.

SNR	No Proc.	STSA-MMSE	NMF	NS DRNN	NG DRNN
-5 dB	32.0	28.0	26.1	17.3	**16.9**
0 dB	27.3	23.5	17.9	14.2	**12.9**
5 dB	21.7	21.4	13.4	**11.1**	11.6
10 dB	16.3	16.7	10.3	**8.67**	11.3
15 dB	11.3	12.6	9.07	**8.75**	10.0
20 dB	8.49	10.9	**7.54**	7.75	9.87
Clean	**6.67**	11.7	8.07	8.94	11.7
Average	17.7	17.8	13.2	**11.0**	12.0

Table B.4: EER for SV system using CAF noise corrupted speech.

SNR	No Proc.	STSA-MMSE	NMF	NS DRNN	NG DRNN
-5 dB	39.9	40.0	36.8	**24.7**	25.6
0 dB	34.0	33.0	29.9	**17.5**	19.2
5 dB	26.7	26.6	22.8	**14.0**	15.1
10 dB	18.8	19.2	18.0	**11.7**	12.1
15 dB	12.8	13.8	14.3	**9.95**	11.2
20 dB	**8.90**	11.1	13.0	9.35	10.6
Clean	**6.67**	11.7	11.7	11.2	11.7
Average	21.1	22.2	20.9	**14.1**	15.1

Table B.5: EER for SV system using PED noise corrupted speech.

SNR	No Proc.	STSA-MMSE	NMF	NS DRNN	NG DRNN
-5 dB	43.1	38.6	40.7	**29.6**	30.3
0 dB	35.6	31.1	32.9	22.0	**20.2**
5 dB	26.3	22.0	24.0	15.4	**13.7**
10 dB	18.3	15.5	17.4	12.6	**10.8**
15 dB	11.9	12.1	12.2	**8.55**	9.56
20 dB	8.57	10.3	10.3	**8.49**	10.9
Clean	**6.67**	11.7	11.3	12.3	11.7
Average	21.5	20.2	21.3	15.6	**15.3**

corporate an SNR estimator, such that the SE front-ends only are used when needed, i.e. for low SNRs.

A somewhat surprising observation is that the NGDRNN front-end in general performs well and not only outperforms the NMF and STSA-MMSE front-ends for the majority of noise types and SNRs, but also the NSDRNN front-ends for several SNRs and noise types.

This is an observation of practical importance, since it shows that using a single DRNN based front-end, which is both text, SNR, male-speaker and

Table B.6: EER for SV system using SSN noise corrupted speech.

SNR	No Proc.	STSA-MMSE	NMF	NS DRNN	NG DRNN
-5 dB	44.4	35.7	37.8	**20.5**	21.6
0 dB	34.9	25.9	26.9	**14.5**	16.0
5 dB	25.5	18.0	17.7	**11.9**	13.2
10 dB	16.1	11.6	12.1	**10.9**	11.4
15 dB	10.3	9.70	**9.51**	10.0	**9.51**
20 dB	**7.40**	10.3	8.17	9.52	9.48
Clean	**6.67**	11.7	10.5	10.3	11.7
Average	20.8	17.6	17.5	**12.5**	13.3

Table B.7: EER for SV system using STR noise corrupted speech.

SNR	No Proc.	STSA-MMSE	NMF	NS DRNN	NG DRNN
-5 dB	41.1	34.6	35.4	**22.3**	24.3
0 dB	33.7	26.5	27.3	17.4	**16.8**
5 dB	26.2	21.5	19.0	14.6	**13.8**
10 dB	18.3	15.7	14.1	12.4	**10.9**
15 dB	12.0	11.9	12.2	9.61	**9.40**
20 dB	9.01	11.1	10.0	9.10	**8.61**
Clean	**6.67**	11.7	10.5	10.7	11.7
Average	21.0	19.0	18.4	13.7	**13.6**

noise type independent eliminates the need for noise type classification and speaker dependent front-ends as would be required by the NMF front-ends.

The advantage of NMF based front-ends is that they can efficiently utilize small amounts of data. In [20] it is shown that using only three utterances from a speaker, a NMF based SE front-end can be designed which outperforms a STSA-MMSE based SE front-end, a Wiener filtering based SE front-end and a spectral subtraction based SE front-end.

Since DNNs typically require a large amount of data, constructing speaker specific front-ends is not practically feasible, since it would require that each SV user should record large amount of enrollment speech. The results presented in Tables B.2 – B.7 show that conventional speech corpora, such as RSR2015, can be used to design a male-speaker and text-independent SE front-end that achieves state-of-the-art performance for a number of noise types and SNRs, hence the NGDRNN front-end can be used for noise robust text-dependent and text-independent speaker verification.

6 Conclusion

In this paper a Deep Recurrent Neural Network (DRNN) based Speech Enhancement (SE) algorithm has been studied in the context of noise-robust text-dependent Speaker Verification (SV). Specifically, a state-of-the-art Long-Short Term Memory (LSTM) based DRNN, trained to be either noise type specific or noise type general as well as text and male-speaker independent is used as denoising front-ends for an i-vector based SV system. Finally, the SV performance of the DRNN based SE front-ends are compared against speaker, text, and noise type dependent Non-negative Matrix Factorization (NMF) based SE front-ends as well as a Short-Time Spectral Amplitude Minimum Mean Square Error (STSA-MMSE) based SE front-end, which is speaker, text and noise type independent.

We show that the noise type specific DRNN based SE front-ends outperform both the NMF based front-ends as well as the STSA-MMSE based front-end for an SNR range from -5 dB – 10 dB, for six different noise types. Furthermore, we show that a text, male-speaker and noise type independent DRNN based SE front-end similarly outperforms both the NMF based SE front-ends and the STSA-MMSE based SE front-end at SNRs below 15 dB. This is a result of great practical importance, since it shows that a single DRNN based SE front-end can achieve state-of-the-art SV performance in a variety of noisy environments, hence eliminating the need for noise type classification and speaker dependent front-ends.

7 Acknowledgment

The authors would like to thank Nicolai Bæk Thomsen for assistance and software used for the speaker verification and non-negative matrix factorization baseline systems. Also, we would like to thank Dong Yu for useful discussions regarding CNTK and DNN training. Finally, we gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan X GPU used for this research.

The paper reflects some results from the OCTAVE Project (#647850), funded by the Research European Agency (REA) of the European Commission, in its framework program Horizon 2020. The views expressed in this paper are those of the authors and do not engage any official position of the European Commission.

References

[1] S. Cumming, "Adoption of Biometric Technologies in Private and Public Sectors Driving Global Markets, Reports BCC Research," *Marketwire*, 2016.

[2] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016, book in preparation for MIT Press. [Online]. Available: http://www.deeplearningbook.org/

[3] K. He, X. Zhang, S. Ren, and J. Sun, "Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification," *arXiv:1502.01852 [cs]*, Feb. 2015.

[4] M. Bojarski *et al.*, "End to End Learning for Self-Driving Cars," *arXiv:1604.07316 [cs]*, Apr. 2016.

[5] G. E. Dahl, D. Yu, L. Deng, and A. Acero, "Context-Dependent Pre-Trained Deep Neural Networks for Large-Vocabulary Speech Recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 1, pp. 30–42, Jan. 2012.

[6] D. Amodei *et al.*, "Deep Speech 2: End-to-End Speech Recognition in English and Mandarin," *arXiv:1512.02595 [cs]*, Dec. 2015.

[7] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "DeepFace: Closing the Gap to Human-Level Performance in Face Verification," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, Jun. 2014, pp. 1701–1708.

[8] D. Yu and L. Deng, *Automatic Speech Recognition: A Deep Learning Approach*, ser. Signals and Communication Technology. London: Springer London, 2015.

[9] A. Hannun *et al.*, "Deep Speech: Scaling up end-to-end speech recognition," *arXiv:1412.5567 [cs]*, Dec. 2014.

[10] X. Zhao, Y. Wang, and D. Wang, "Robust speaker identification in noisy and reverberant conditions," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, May 2014, pp. 3997–4001.

[11] C. A. Medina, A. Alcaim, and J. A. Apolinario, "Wavelet denoising of speech using neural networks for threshold selection," *Electron. Lett.*, vol. 39, no. 25, pp. 1869–1871, Dec. 2003.

[12] A. A. Nugraha, K. Yamamoto, and S. Nakagawa, "Single-channel dereverberation by feature mapping using cascade neural networks for robust distant speaker identification and speech recognition," *EURASIP J. Audio Speech Music Process.*, p. 13, Apr. 2014.

[13] S. Du, X. Xiao, and E. S. Chng, "DNN feature compensation for noise robust speaker verification," in *2015 IEEE China Summit and Int. Conf. on Sig. and Inf. Proc. (ChinaSIP)*, Jul. 2015, pp. 871–875.

[14] J.-C. Wang, C.-H. Yang, J.-F. Wang, and H.-P. Lee, "Robust speaker identification and verification," *IEEE Comput. Intell. Mag.*, vol. 2, no. 2, pp. 52–59, May 2007.

[15] S. Omid Sadjadi and J. H. L. Hansen, "Assessment of Single-Channel Speech Enhancement Techniques for Speaker Identification Under Mismatched Conditions," in *11th Annual Conference of the International Speech Communication Association*, 2010.

[16] A. Moreno-Daniel, J. A. Nolazco-Flores, T. Wada, and B. H. Juang, "Acoustic Model Enhancement: An Adaptation Technique for Speaker Verification Under Noisy Environments," in *2007 IEEE International Conference on Acoustics, Speech and Signal Processing - ICASSP '07*, vol. 4, Apr. 2007, pp. IV–289–IV–292.

[17] C. A. Medina, J. A. Apolinario, A. Alcaim, and R. G. Alves, "Robust speaker verification in colored noise environment," in *Conference Record of the Thirty-Seventh Asilomar Conference on Signals, Systems and Computers, 2004*, vol. 2, Nov. 2003, pp. 1890–1893.

[18] J. A. Nolazco-Flores and L. P. Garcia-Perera, "Enhancing acoustic models for robust speaker verification," in *2008 IEEE International Conference on Acoustics, Speech and Signal Processing*, Mar. 2008, pp. 4837–4840.

[19] J. Ming, T. J. Hazen, and J. R. Glass, "Speaker Verification Over Handheld Devices with Realistic Noisy Speech Data," in *2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings*, vol. 1, May 2006.

[20] N. Bæk Thomsen, D. Alexander Lehmann Thomsen, Z.-H. Tan, B. Lindberg, and S. Holdt Jensen, "Speaker-Dependent Dictionary-based Speech Enhancement for Text-Dependent Speaker Verification," in *INTERSPEECH*, 2016.

[21] D. Garcia-Romero, X. Zhou, and C. Y. Espy-Wilson, "Multicondition training of Gaussian PLDA models in i-vector space for noise and reverberation robust speaker recognition," in *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Mar. 2012, pp. 4257–4260.

[22] J. Villalba and E. Lleida, "Handling i-vectors from different recording conditions using multi-channel simplified PLDA in speaker recognition," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, May 2013, pp. 6763–6767.

[23] J. Ming, T. J. Hazen, J. R. Glass, and D. A. Reynolds, "Robust Speaker Recognition in Noisy Conditions," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 5, pp. 1711–1723, Jul. 2007.

[24] T. T. Dat, J. Y. Kim, H. G. Kim, and K. R. Lee, "Robust Speaker Verification Using Low-Rank Recovery under Total Variability Space," in *2015 5th International Conference on IT Convergence and Security (ICITCS)*, Aug. 2015, pp. 1–4.

[25] R. Saeidi *et al.*, "INTERSPEECH 2013 I4u submission to NIST SRE 2012: A large-scale collaborative effort for noise-robust speaker verification," in *INTERSPEECH*, 2013.

[26] S. E. Shepstone, K. A. Lee, H. Li, Z. H. Tan, and S. H. Jensen, "Total Variability Modeling Using Source-Specific Priors," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 3, pp. 504–517, Mar. 2016.

[27] J. H. L. Hansen and T. Hasan, "Speaker Recognition by Machines and Humans: A tutorial review," *IEEE Signal Processing Magazine*, vol. 32, no. 6, pp. 74–99, Nov. 2015.

[28] A. Hurmalainen, R. Saeidi, and T. Virtanen, "Noise Robust Speaker Recognition with Convolutive Sparse Coding," in *INTERSPEECH 2015, 16th Annual Conference of the International Speech Communication Association*, 2015, pp. 244 – 248.

[29] ——, "Exemplar-based sparse representation and sparse discrimination for noise robust speaker identification," in *Odyssey 2012: The Speaker and Language Recognition Workshop*, Singapore, 2012.

[30] ——, "Similarity induced group sparsity for non-negative matrix factorisation," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Apr. 2015, pp. 4425–4429.

[31] P. C. Loizou, *Speech Enhancement: Theory and Practice*. CRC Press, 2013, vol. 2013.

[32] J. Erkelens, R. Hendriks, R. Heusdens, and J. Jensen, "Minimum Mean-Square Error Estimation of Discrete Fourier Coefficients With Generalized Gamma Priors," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 15, no. 6, pp. 1741–1752, Aug. 2007.

[33] Y. Wang, "Supervised Speech Separation Using Deep Neural Networks," Ph.D. dissertation, The Ohio State University, 2015.

[34] D. Williamson, Y. Wang, and D. Wang, "Deep neural networks for estimating speech model activations," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, Apr. 2015, pp. 5113–5117.

[35] D. S. Williamson, Y. Wang, and D. Wang, "Estimating nonnegative matrix model activations with deep neural networks to increase perceptual speech quality," *J. Acoust. Soc. Am.*, vol. 138, no. 3, pp. 1399–1407, Sep. 2015.

[36] A. Larcher, K. A. Lee, B. Ma, and H. Li, "Text-dependent speaker verification: Classifiers, databases and RSR2015," *Speech Communication*, vol. 60, pp. 56–77, May 2014.

[37] J. Barker, M. Ricard, V. Vincent, and S. Watanabe, "The third 'CHiME' Speech Separation and Recognition Challenge: Dataset, task and baselines," *ASRU*, 2015.

[38] J. Chen, Y. Wang, S. E. Yoho, D. Wang, and E. W. Healy, "Large-scale training to increase speech intelligibility for hearing-impaired listeners in novel noises," *The Journal of the Acoustical Society of America*, vol. 139, no. 5, pp. 2604–2612, May 2016.

[39] P.-S. Huang, M. Kim, M. Hasegawa-Johnson, and P. Smaragdis, "Joint Optimization of Masks and Deep Recurrent Neural Networks for Monaural Source Separation," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 23, no. 12, pp. 2136–2147, Dec. 2015.

[40] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997.

[41] E. W. Healy, S. E. Yoho, J. Chen, Y. Wang, and D. Wang, "An algorithm to increase speech intelligibility for hearing-impaired listeners in novel segments of the same noise type," *J. Acoust. Soc. Am.*, vol. 138, no. 3, pp. 1660–1669, Sep. 2015.

[42] Y. Wang, A. Narayanan, and D. Wang, "On Training Targets for Supervised Speech Separation," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 22, no. 12, pp. 1849–1858, Dec. 2014.

[43] D. Yu, M. Kolbæk, Z.-H. Tan, and J. Jensen, "Permutation Invariant Training of Deep Models for Speaker-Independent Multi-talker Speech Separation," *arXiv:1607.00325 [cs]*, Jul. 2016.

[44] A. Agarwal *et al.*, "An introduction to computational networks and the computational network toolkit," Microsoft Technical Report {MSR-TR}-2014-112, Tech. Rep., 2014.

[45] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Improving neural networks by preventing co-adaptation of feature detectors," *arXiv: 1207.0580*, Jul. 2012.

[46] D. D. Lee and H. S. Seung, "Algorithms for Non-negative Matrix Factorization," *Neural Information Processing Systems - NIPS*, vol. 13, pp. 556–562, 2000.

[47] M. N. Schmidt, J. Larsen, and F. T. Hsiao, "Wind Noise Reduction using Non-Negative Sparse Coding," in *2007 IEEE Workshop on Machine Learning for Signal Processing*, Aug. 2007, pp. 431–436.

[48] R. C. Hendriks, T. Gerkmann, and J. Jensen, "DFT-Domain Based Single-Microphone Noise Reduction for Speech Enhancement: A Survey of the State of the Art," *Synthesis Lectures on Speech and Audio Processing*, vol. 9, no. 1, pp. 1–80, Jan. 2013.

[49] Y. Ephraim and D. Malah, "Speech enhancement using a minimum-mean square error short-time spectral amplitude estimator," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 32, no. 6, pp. 1109–1121, Dec. 1984.

[50] R. Hendriks, R. Heusdens, and J. Jensen, "MMSE based noise PSD tracking with low complexity," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, Mar. 2010, pp. 4266–4269.

[51] N. Dehak, P. J. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-End Factor Analysis for Speaker Verification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 788–798, May 2011.

[52] D. Povey *et al.*, "The kaldi speech recognition toolkit," in *In IEEE 2011 workshop*, 2011.

[53] J. S. Garofolo, L. F. Lamel, W. M. Fisher, J. G. Fiscus, D. S. Pallett, and N. L. Dahlgren, "DARPA TIMIT Acoustic Phonetic Continuous Speech Corpus CDROM," 1993.

This page intentionally left blank.

Paper C

Permutation Invariant Training of Deep Models for Speaker-Independent Multi-Talker Speech Separation

Dong Yu, Morten Kolbæk, Zheng-Hua Tan, and Jesper Jensen

The paper has been published in
Proceedings IEEE International Conference on Acoustics, Speech, and Signal Processing, pp. 241-245, 2017.

© 2017 IEEE
The layout has been revised.

Abstract

We propose a novel deep learning training criterion, named Permutation Invariant Training (PIT), for speaker independent multi-talker speech separation, commonly known as the cocktail-party problem. Different from the multi-class regression technique and the Deep Clustering (DPCL) technique, our novel approach minimizes the separation error directly. This strategy effectively solves the long-lasting label permutation problem, that has prevented progress on deep learning based techniques for speech separation. We evaluated PIT on the WSJ0 and Danish mixed-speech separation tasks and found that it compares favorably to Non-negative Matrix Factorization (NMF), Computational Auditory Scene Analysis (CASA), and DPCL and generalizes well over unseen speakers and languages. Since PIT is simple to implement and can be easily integrated and combined with other advanced techniques, we believe improvements built upon PIT can eventually solve the cocktail-party problem.

1 Introduction

Despite the significant progress made in dictating single-speaker speech in the recent years [1–4], the progress made in multi-talker mixed speech separation and recognition, often referred to as the cocktail-party problem [5, 6], has been less impressive. Although human listeners can easily perceive separate sources in an acoustic mixture, the same task seems to be extremely difficult for automatic computing systems, especially when only a single microphone recording of the mixed-speech is available [7, 8].

Nevertheless, solving the cocktail-party problem is critical to enable scenarios such as automatic meeting transcription, automatic captioning for audio/video recordings (e.g., YouTube), and multi-party human-machine interactions (e.g., in the world of Internet of things (IoT)), where speech overlapping is commonly observed.

Over the decades, many attempts have been made to attack this problem. Before the deep learning era, the most popular technique was Computational Auditory Scene Analysis (CASA) [9, 10]. In this approach, certain segmentation rules based on perceptual grouping cues [11] are (often semi-manually) designed to operate on low-level features to estimate a time-frequency mask that isolates the signal components belonging to different speakers. This mask is then used to reconstruct the signal. Non-negative Matrix Factorization (NMF) [12–14] is another popular technique which aims to learn a set of non-negative bases that can be used to estimate mixing factors during evaluation. Both CASA and NMF led to very limited success in separating sources in multi-talker mixed speech [7]. The most successful technique before the deep learning era is the model based approach [15–17], such as fac-

torial GMM-HMM [18], that models the interaction between the target and competing speech signals and their temporal dynamics. Unfortunately this model assumes and only works under closed-set speaker condition.

Motivated by the success of deep learning techniques in single-talker Automatic Speech Recognition (ASR) [1–4], researchers have developed many deep learning techniques for speech separation in recent years. Typically, networks are trained based on parallel sets of mixtures and their constituent target sources [19–22]. The networks are optimized to predict the source belonging to the target class, usually for each time-frequency bin. Unfortunately, these works often focus on, and only work for, separating speech from (often challenging) background noise (or music) because speech has very different characteristics than noise/music. Note that there are indeed works that are aiming at separating multi-talker mixed speech (e.g. [22]). However, these works rely on speaker-dependent models by assuming that the (often few) target speakers are known during training.

The difficulty in speaker-independent multi-talker speech separation comes from the label ambiguity or permutation problem (which will be described in Section 2). Only two deep learning based works [8, 23, 24] have tried to address and solve this harder problem. In Weng et al. [8], which achieved the best result on the dataset used in 2006 monaural speech separation and recognition challenge [7], the instantaneous energy was used to solve the label ambiguity problem and a two-speaker joint-decoder with speaker switching penalty was used to separate and trace speakers. This approach tightly couples with the decoder and is difficult to scale up to more than two speakers due to the way labels are determined. Hershey et al. [23, 24] made significant progress with their Deep Clustering (DPCL) technique. In their work, they trained an embedding for each time-frequency bin to optimize a segmentation (clustering) criterion. During evaluation, each time-frequency bin was first mapped into the embedding space upon which a clustering algorithm was used to generate a partition of the time-frequency bins. Impressively, their systems trained on two-talker mixed-speech perform well on three-talker mixed-speech. However, in their approach it is assumed that each time-frequency bin belongs to only one speaker (i.e., a partition) due to the clustering step. Although this is often a good approximation, it is known to be sub-optimal. Furthermore, their approach is hard to combine with other techniques such as complex-domain separation.

In this paper, we propose a novel training criterion, named Permutation Invariant Training (PIT), for speaker independent multi-talker speech separation. Most prior arts treat speech separation as either a multi-class regression problem or a segmentation (or clustering) problem. PIT, however, considers it a *separation* problem (as it should be) by minimizing the separation error. More specifically, PIT first determines the best output-target assignment and then minimizes the error given the assignment. This strategy, which

is directly implemented inside the network structure, elegantly solves the long-lasting label permutation problem that has prevented progress on deep learning based techniques for speech separation.

We evaluated PIT on the WSJ0 and Danish mixed-speech separation tasks. Experimental results indicate that PIT compares favorably to NMF, CASA, and DPCL and generalizes well over unseen speakers and languages. In other words, through the training process PIT learns acoustic cues for source separation, which are both speaker and language independent, similar to humans. Since PIT is simple to implement and can be easily integrated and combined with other advanced techniques we believe improvements built upon PIT can eventually solve the cocktail-party problem.

2 Monaural Speech Separation

The goal of monaural speech separation is to estimate the individual source signals in a linearly mixed, single-microphone signal, in which the source signals overlap in the time-frequency domain. Let us denote the S source signal sequences in the time domain as $\mathbf{x}_s(t), s = 1, \cdots, S$ and the mixed signal sequence as $\mathbf{y}(t) = \sum_{s=1}^{S} \mathbf{x}_s(t)$. The corresponding Short-Time Fourier Transform (STFT) of these signals are $\mathbf{X}_s(t,f)$ and $\mathbf{Y}(t,f) = \sum_{s=1}^{S} \mathbf{X}_s(t,f)$, respectively, for each time t and frequency f. Given $\mathbf{Y}(t,f)$, the goal of monaural speech separation is to recover each source $\mathbf{X}_s(t,f)$.

In a typical setup, it is assumed that only STFT magnitude spectra is available. The phase information is ignored during the separation process and is used only when recovering the time domain waveforms of the sources.

Obviously, given only the magnitude of the mixed spectrum $|\mathbf{Y}(t,f)|$, the problem of recovering $|\mathbf{X}_s(t,f)|$ is ill-posed, as there are an infinite number of possible $|\mathbf{X}_s(t,f)|$ combinations that lead to the same $|\mathbf{Y}(t,f)|$. To overcome this core problem, the system has to learn from some training set $\mathbb{S}$ that contains pairs of $|\mathbf{Y}(t,f)|$ and $|\mathbf{X}_s(t,f)|$ to look for regularities. More specifically, we train a deep learning model $g(\cdot)$ such that $g\left(f(|\mathbf{Y}|);\theta\right) = |\tilde{\mathbf{X}}_s|, s = 1, \cdots, S$, where θ is a model parameter vector, and $f(|\mathbf{Y}|)$ is some feature representation of $|\mathbf{Y}|$. For simplicity and clarity we have omitted, and will continue to omit, time-frequency indexes when there is no ambiguity.

It is well-known (e.g., [19]) that better results can be achieved if, instead of estimating $|\mathbf{X}_s|$ directly, we first estimate a set of masks $\mathbf{M}_s(t,f)$ using a deep learning model $h\left(f(|\mathbf{Y}|);\theta\right) = \tilde{\mathbf{M}}_s(t,f)$ with the constraint that $\tilde{\mathbf{M}}_s(t,f) \geq 0$ and $\sum_{s=1}^{S} \tilde{\mathbf{M}}_s(t,f) = 1$ for all time-frequency bins (t,f). This constraint can be easily satisfied with the softmax operation. We then estimate $|\mathbf{X}_s|$ as $|\tilde{\mathbf{X}}_s| = \tilde{\mathbf{M}}_s \circ |\mathbf{Y}|$, where $\circ$ is the element-wise product of two operands. This strategy is adopted in this study.

Note that since we first estimate masks, the model parameters can be optimized to minimize the Mean Squared Error (MSE) between the estimated mask $\tilde{\mathbf{M}}_s$ and the Ideal Ratio Mask (IRM) $\mathbf{M}_s = \frac{|\mathbf{X}_s|}{|\mathbf{Y}|}$,

$$J_m = \frac{1}{T \times F \times S} \sum_{s=1}^{S} \|\tilde{\mathbf{M}}_s - \mathbf{M}_s\|^2,$$

where T and F denote the number of time frames and frequency bins, respectively. This approach comes with two problems. First, in silence segments, $|\mathbf{X}_s| = 0$ and $|\mathbf{Y}| = 0$, so that $\mathbf{M}_s$ is not well defined. Second, what we really care about is the error between the estimated magnitude and the true magnitude of each source, while a smaller error on masks may not lead to a smaller error on magnitude.

To overcome these limitations, recent works [19] directly minimize the MSE

$$J_x = \frac{1}{T \times F \times S} \sum_{s=1}^{S} \||\tilde{\mathbf{X}}_s| - |\mathbf{X}_s|\|^2$$

between the estimated magnitude and the true magnitude. Note that in silence segments $|\mathbf{X}_s| = 0$ and $|\mathbf{Y}| = 0$, and so the accuracy of mask estimation does not affect the training criterion for those segments. In this study, we estimate masks $\tilde{\mathbf{M}}_s$ which minimize J_x.

3 Permutation Invariant Training

Except DPCL [23, 24], all other recent speech separation works treat the separation problem as a multi-class regression problem. In their architecture, N frames of feature vectors of the mixed signal $|\mathbf{Y}|$ are used as the input to deep learning models, such as Deep Neural Networks (DNNs), Convolutional Neural Networks (CNNs), and Long Short-Term Memory (LSTM) Recurrent Neural Networks (RNNs), to generate one (often the center) frame of masks for each talker. These masks are then used to construct one frame of single-source speech $|\tilde{\mathbf{X}}_1|$ and $|\tilde{\mathbf{X}}_2|$, for source 1 and 2, respectively.

During training we need to provide the correct reference (or target) magnitude $|\mathbf{X}_1|$ and $|\mathbf{X}_2|$ to the corresponding output layers for supervision. Since the model has multiple output layers, one for each mixing source, and they depend on the same input mixture, reference assigning can be tricky especially if the training set contains many utterances spoken by many speakers. This problem is referred to as the label ambiguity (or permutation) problem in [8, 23]. Due to this problem, prior arts perform poorly on speaker-independent multi-talker speech separation. It was believed that speaker-independent multi-talker speech separation is not feasible [25].

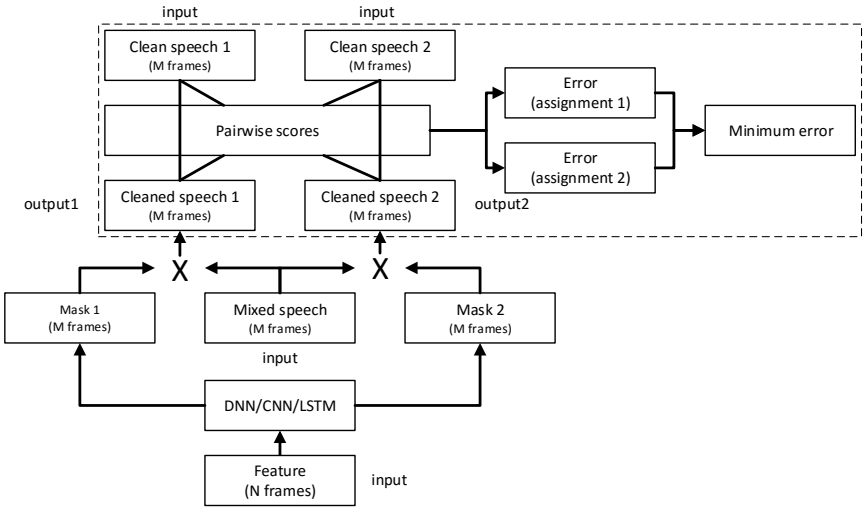


Fig. C.1: The two-talker speech separation model with permutation invariant training.

The solution proposed in this work is illustrated in Figure C.1. There are two key inventions in this novel model: permutation invariant training and segment-based decision making.

In our new model the reference source streams are given as a set instead of an ordered list. In other words, the same training result is obtained, no matter in which order these sources are listed. This behavior is achieved with PIT highlighted inside the dashed rectangular in Figure C.1. In order to associate references to the output layers, we first determine the (total number of $S!$) possible assignments between the references and the estimated sources. We then compute the total MSE for each assignment, which is defined as the combined pairwise MSE between each reference $|\mathbf{X}_s|$ and the estimated source $|\tilde{\mathbf{X}}_s|$. The assignment with the least total MSE is chosen and the model is optimized to reduce this particular MSE. In other words we simultaneously conduct label assignment and error evaluation. Similar to the prior arts, PIT uses as input N successive frames (i.e., an input *meta-frame*) of features to exploit the contextual information. Different from the prior arts, the output of the PIT is also a window of frames. With PIT, we directly minimize the separation error at the meta-frame level. Although the number of speaker assignments is factorial in the number of speakers, the pairwise MSE computation is only quadratic, and more importantly the MSE computation can be completely ignored during evaluation.

During inference, the only information available is the mixed speech. Speech separation can be directly carried out for each input meta-frame, for

which an output meta-frame with M frames of speech is estimated for each stream. The input meta-frame is then shifted by one or more frames. Due to the PIT training criterion, output-to-speaker assignment may change across frames. In the simplest setup, we can just assume they do not change when reconstructing sources. Better performance may be achieved if a speaker-tracing algorithm is applied on top of the output of the network.

Once the relationship between the outputs and source streams are determined for each output meta-frame, the separated speech can be estimated, taking into account all meta-frames by, for example, averaging the same frame across meta-frames.

4 Experimental Results

4.1 Datasets

We evaluated PIT on the WSJ0-2mix and Danish-2mix datasets. The WSJ0-2mix dataset was introduced in [23] and was derived from WSJ0 corpus [26]. The 30h training set and the 10h validation set contains two-speaker mixtures generated by randomly selecting speakers and utterances from the WSJ0 training set si_tr_s, and mixing them at various signal-to-noise ratios (SNRs) uniformly chosen between 0 dB and 5 dB. The 5h test set was similarly generated using utterances from 16 speakers from the WSJ0 validation set si_dt_05 and evaluation set si_et_05.

The Danish-2mix dataset was constructed from the Danish corpus [27], which consists of approximately 560 speakers each speaking 312 utterances with average utterance duration of approximately 5 sec. The dataset was constructed by randomly selecting a set of 45 male and 45 female speakers from the corpus, and then allocating 232, 40, and 40 utterances from each speaker to generate mixed speech in the training, validation and Closed-Condition (CC) (seen speaker) test set, respectively. 40 utterances from each of another 45 male and 45 female speakers were randomly selected to construct the Open-Condition (OC) (unseen speaker) test set. Speech mixtures were constructed in the way similar to the WSJ0-2mix dataset, but all mixed with 0 dB - the hardest condition. We constructed 10k and 1k mixtures in total in the training and validation set, respectively, and 1k mixtures for each of the CC and OC test sets. The Danish-3mix (three-talker mixed speech) dataset was constructed similarly.

In this study we focus on the WSJ0-2mix dataset so that we can directly compare PIT with published state-of-the-art results obtained using other techniques.

4.2 Models

Our models were implemented using the Microsoft Cognitive Toolkit (CNTK) [28]. The feed-forward DNN (denoted as DNN) has three hidden layers each with 1024 ReLU units. In (inChannel, outChannel)-(strideW, strideH) format, the CNN model has one $(1,64)-(2,2)$, four $(64,64)-(1,1)$, one $(64,128)-(2,2)$, two $(128,128)-(1,1)$, one $(128,256)-(2,2)$, and two $(256,256)-(1,1)$ convolution layers with 3×3 kernels, a pooling layer and a 1024-unit ReLU layer. The input to the models is the stack (over multiple frames) of the 257-dim STFT spectral magnitude of the speech mixture, computed using STFT with a frame size of 32ms and 16ms shift. There are S output streams for S-talker mixed speech. Each output stream has a dimension of $257 \times M$, where M is the number of frames in the output meta-frame. In our study, the validation set is only used to control the learning rate.

4.3 Training Behavior

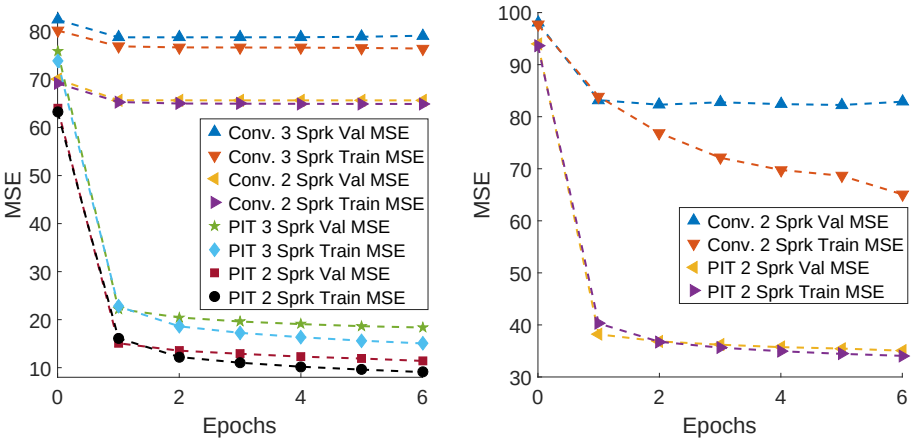


Fig. C.2: MSE over epochs on the Danish (left) and WSJ0 (right) training and validation sets with conventional training and PIT.

In Figure C.2 we plotted the DNN training progress as measured by the MSE on the training and validation set with conventional training and PIT on the mixed speech datasets described in subsection 4.1. From the figure we can see clearly that the validation MSE hardly decreases with the conventional approach due to the label permutation problem discussed in [8, 23]. In contrast, training converges quickly to a much better MSE for both two- and three-talker mixed speech when PIT is used.

Table C.1: SDR improvements (dB) for different separation methods on the WSJ0-2mix dataset.

Method	Input\Output window	Opt. Assign CC	Opt. Assign OC	Def. Assign CC	Def. Assign OC
Oracle NMF [23]	-	-	-	5.1	-
CASA [23]	-	-	-	2.9	3.1
DPCL [23]	100\100	6.5	6.5	5.9	5.8
DPCL+ [24]	100\100	-	-	-	10.3
PIT-DNN	101\101	6.2	6.0	5.3	5.2
PIT-DNN	51\51	7.3	7.2	**5.7**	**5.6**
PIT-DNN	41\7	10.1	10.0	-0.3	-0.6
PIT-DNN	41\5	**10.5**	**10.4**	-0.6	-0.8
PIT-CNN	101\101	8.4	8.6	**7.7**	**7.8**
PIT-CNN	51\51	9.6	9.7	7.5	7.7
PIT-CNN	41\7	10.7	10.7	-0.6	-0.7
PIT-CNN	41\5	**10.9**	**10.9**	-0.8	-0.9
IRM	-	12.3	12.5	12.3	12.5

4.4 Signal-to-Distortion Ratio Improvement

We evaluated PIT on its potential to improve the Source-to-Distortion Ratio (SDR) [29], a metric widely used to evaluate speech enhancement performance.

In Table C.1 we summarized the SDR improvement in dB from different separation configurations for two-talker mixed speech in CC and OC. In these experiments each frame was reconstructed by averaging over all output meta-frames that contain the same frame. In the default assignment setup it is assumed that there is no output-speaker switch across frames (which is not true). This is the improvement achievable using PIT without any speaker tracing. In the optimal assignment setup, the output-speaker assignment for each output meta-frame is determined based on mixing streams. This reflects the separation performance within each segment (meta-frame) and is the improvement achievable when the speakers are correctly traced. The gap between these two values indicates the contribution from speaker tracing. As a reference, we also provided the IRM result which is the oracle and upper bound achievable on this task.

From the table we can make several observations. First, without speaker tracing (def. assign) PIT can achieve similar and better performance than the original DPCL [23], respectively, with DNN and CNN, but under-performs the more complicated DPCL+ [24]. Note that, PIT is much simpler than even the original (simpler) DPCL and we did not fine-tune architectures and learning procedures as done in [24]. Second, as we reduce the output window size we can improve the separation performance within each window and achieve better SDR improvement if speakers are correctly traced (opt. assign). However, when output window size is reduced, the output-speaker assignment

Table C.2: SDR improvements (dB) based on optimal assignment for DNNs trained with Danish-2mix.

Method	Input\Output window	CC	OC	WSJ0 OC
IRM	-	17.2	17.3	13.2
PIT-DNN	101\101	9.00	8.61	4.29
PIT-DNN	61\61	9.87	9.44	5.17
PIT-DNN	31\31	11.1	10.7	6.18
PIT-DNN	31\7	14.0	13.8	9.03
PIT-DNN	31\5	14.1	13.9	9.29

changes more frequently as indicated by the poor default assignment performance. Speaker tracing thus becomes more important given the larger gap between the opt. assign and def. assign. Fourth, PIT generalizes well on unseen speakers since the performances on the open and closed conditions are very close. Fifth, powerful models such as CNN consistently outperforms DNNs but the gain diminishes when the output window size is small.

In Table C.2 we summarized the SDR improvement in dB with optimal assignment from different configurations for DNNs trained on Danish-2mix. We also report SDR improvement using a dataset constructed identical to Danish-2mix but based on the si_tr_s data from WSJ0. Besides the findings obtained in Table C.1, an interesting observation is that although the system has never seen English speech, it performs remarkably well on this WSJ0 dataset when compared to the IRM (oracle) values. These results indicate that the separation ability learned with PIT generalizes well not only across speakers but also across languages.

5 Conclusion and Discussion

In this paper, we have described a novel permutation invariant training technique for speaker-independent multi-talker speech separation. To the best of our knowledge this is the first successful work that employs the separation view (and criterion) of the task[1], instead of the multi-class regression or segmentation view that are used in prior arts. This is a big step towards solving the important cocktail-party problem in a real-world setup, where the set of speakers are unknown during the training time.

Our experiments on two-talker mixed speech separation tasks demonstrate that PIT trained models generalize well to unseen speakers and languages. Although our results are mainly on two-talker separation tasks, PIT

[1]Hershey et al. [23] tried PIT (called permutation free training in their paper) but failed to make it work. They retried after reading the preprint of this work and now got positive results as well.

can be easily and effectively extended to the three-talker case as shown in figure C.2.

In this paper we focused on PIT - the key technique that enables training for the separation of multi-talker mixed speech. PIT is much simpler yet performs better than the original DPCL [23] that contains separate embedding and clustering stages.

Since PIT, as a training technique, can be easily integrated and combined with other advanced techniques, it has great potential for further improvement. We believe improvements can come from work in the following areas:

First, due to the change of output-speaker assignment across frames, there is a big performance gap between the optimal output-speaker assignment and the default assignment, especially in the same-gender case and when the output window size is small. This gap can be reduced with separate speaker tracing algorithms that exploit the overlapping frames and speaker characteristics (e.g., similarity) in output meta-frames. It is also possible to train an end-to-end system in which speaker tracing is directly built into the model, e.g., by applying PIT at utterance level. We will report these results in other papers.

Second, we only explored simple DNN/CNN structures in this work. More powerful models such as bi-directional LSTMs, CNNs with deconvolution layers, or even just larger models may further improve the performance. Hyper-parameter tuning will also help and sometimes lead to significant performance gain.

Third, in this work we reconstructed source streams from spectral magnitude only. Unlike DPCL, PIT can be easily combined with reconstruction techniques that exploit complex-valued spectrum to further boost performance.

Fourth, the acoustic cues learned by the model are largely speaker and language independent. It is thus possible to train a universal speech separation model using speech in various speakers, languages, and noise conditions.

Finally, although we focused on monaural speech separation in this work, the same technique can be deployed in the multi-channel setup and combined with techniques such as beamforming due to its flexibility. In fact, since beamforming and PIT separate speech using different information, they complement with each other. For example, speaker tracing may be much easier when beamforming is available.

6 Acknowledgment

We thank Dr. John Hershey at MERL and Zhuo Chen at Columbia University for sharing the WSJ0-2mix data list and for valuable discussions.

References

[1] D. Yu, L. Deng, and G. E. Dahl, "Roles of pre-training and fine-tuning in context-dependent dbn-hmms for real-world speech recognition," in *NIPS 2010 Workshop on Deep Learning and Unsupervised Feature Learning*, 2010.

[2] G. E. Dahl, D. Yu, L. Deng, and A. Acero, "Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 20, no. 1, pp. 30–42, 2012.

[3] F. Seide, G. Li, and D. Yu, "Conversational speech transcription using context-dependent deep neural networks," in *INTERSPEECH*, 2011, pp. 437–440.

[4] G. Hinton *et al.*, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 82–97, 2012.

[5] E. C. Cherry, "Some experiments on the recognition of speech, with one and with two ears," *The Journal of the Acoustical Society of America*, vol. 25, no. 5, pp. 975–979, 1953.

[6] A. S. Bregman, *Auditory scene analysis: The perceptual organization of sound*. MIT press, 1994.

[7] M. Cooke, J. R. Hershey, and S. J. Rennie, "Monaural speech separation and recognition challenge," *Computer Speech and Language*, vol. 24, no. 1, pp. 1–15, 2010.

[8] C. Weng, D. Yu, M. L. Seltzer, and J. Droppo, "Deep neural networks for single-channel multi-talker speech recognition," *IEEE/ACM Transactions on Audio, Speech and Language Processing*, vol. 23, no. 10, pp. 1670–1679, 2015.

[9] M. Cooke, *Modelling auditory processing and organisation*. Cambridge University Press, 2005, vol. 7.

[10] D. P. Ellis, "Prediction-driven computational auditory scene analysis," Ph.D. dissertation, Massachusetts Institute of Technology, 1996.

[11] M. Wertheimer, *Laws of organization in perceptual forms*. Kegan Paul, Trench, Trubner & Company, 1938.

[12] M. N. Schmidt and R. K. Olsson, "Single-channel speech separation using sparse non-negative matrix factorization," in *INTERSPEECH*, 2006.

[13] P. Smaragdis, "Convolutive speech bases and their application to supervised speech separation," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 15, no. 1, pp. 1–12, 2007.

[14] J. Le Roux, F. Weninger, and J. Hershey, "Sparse nmf–half-baked or well done," *Mitsubishi Electric Research Labs (MERL), Cambridge, MA, USA, Tech. Rep., no. TR2015-023*, 2015.

[15] T. T. Kristjansson, J. R. Hershey, P. A. Olsen, S. J. Rennie, and R. A. Gopinath, "Super-human multi-talker speech recognition: the ibm 2006 speech separation challenge system," in *INTERSPEECH*, vol. 12, 2006, p. 155.

[16] T. Virtanen, "Speech recognition using factorial hidden markov models for separation in the feature space," in *INTERSPEECH*. Citeseer, 2006.

[17] R. J. Weiss and D. P. Ellis, "Monaural speech separation using source-adapted models," in *Applications of Signal Processing to Audio and Acoustics, 2007 IEEE Workshop on*. IEEE, 2007, pp. 114–117.

[18] Z. Ghahramani and M. I. Jordan, "Factorial hidden markov models," *Machine learning*, vol. 29, no. 2-3, pp. 245–273, 1997.

[19] Y. Wang, A. Narayanan, and D. Wang, "On training targets for supervised speech separation," *IEEE/ACM Transactions on Audio, Speech and Language Processing*, vol. 22, no. 12, pp. 1849–1858, 2014.

[20] Y. Xu, J. Du, L.-R. Dai, and C.-H. Lee, "An experimental study on speech enhancement based on deep neural networks," *IEEE Signal Processing Letters*, vol. 21, no. 1, pp. 65–68, 2014.

[21] F. Weninger *et al.*, "Speech enhancement with lstm recurrent neural networks and its application to noise-robust asr," in *Latent Variable Analysis and Signal Separation*. Springer, 2015, pp. 91–99.

[22] P.-S. Huang, M. Kim, M. Hasegawa-Johnson, and P. Smaragdis, "Joint optimization of masks and deep recurrent neural networks for monaural source separation," *IEEE/ACM Transactions on Audio, Speech and Language Processing*, vol. 23, no. 12, pp. 2136–2147, 2015.

[23] J. R. Hershey, Z. Chen, J. L. Roux, and S. Watanabe, "Deep clustering: Discriminative embeddings for segmentation and separation," *arXiv preprint arXiv:1508.04306*, 2015.

[24] Y. Isik, J. L. Roux, Z. Chen, S. Watanabe, and J. R. Hershey, "Single-channel multi-speaker separation using deep clustering," *arXiv preprint arXiv:1607.02173*, 2016.

[25] D. Wang, "Tutorial: Supervised speech separation," in *ICASSP*, 2016.

[26] Garofolo, John, et al., "CSR-I (WSJ0) Complete LDC93S6A," philadelphia: Linguistic Data Consortium, 1993.

[27] Nordisk spraakteknologi holding AS (NST), "Akustiske databaser for dansk," http://www.nb.no/sbfil/dok/nst_taledat_dk.pdf, accessed: 2016-05-19.

[28] A. et al., "An introduction to computational networks and the computational network toolkit," Microsoft Technical Report MSR-TR-2014-112, Tech. Rep., 2014.

[29] E. Vincent, R. Gribonval, and C. Févotte, "Performance measurement in blind audio source separation," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 4, pp. 1462–1469, Jul. 2006.

Paper D

Multi-Talker Speech Separation With Utterance-Level Permutation Invariant Training of Deep Recurrent Neural Networks

Morten Kolbæk, Dong Yu, Zheng-Hua Tan, and Jesper Jensen

The paper has been published in
IEEE/ACM Transactions on Audio, Speech, and Language Processing,
vol. 25, no. 10, pp. 1901-1913, March 2017.

© 2017 IEEE
The layout has been revised.

Abstract

In this paper we propose the utterance-level Permutation Invariant Training (uPIT) technique. uPIT is a practically applicable, end-to-end, deep learning based solution for speaker independent multi-talker speech separation. Specifically, uPIT extends the recently proposed Permutation Invariant Training (PIT) technique with an utterance-level cost function, hence eliminating the need for solving an additional permutation problem during inference, which is otherwise required by frame-level PIT. We achieve this using Recurrent Neural Networks (RNNs) that, during training, minimize the utterance-level separation error, hence forcing separated frames belonging to the same speaker to be aligned to the same output stream. In practice, this allows RNNs, trained with uPIT, to separate multi-talker mixed speech without any prior knowledge of signal duration, number of speakers, speaker identity or gender.

We evaluated uPIT on the WSJ0 and Danish two- and three-talker mixed-speech separation tasks and found that uPIT outperforms techniques based on Non-negative Matrix Factorization (NMF) and Computational Auditory Scene Analysis (CASA), and compares favorably with Deep Clustering (DPCL) and the Deep Attractor Network (DANet). Furthermore, we found that models trained with uPIT generalize well to unseen speakers and languages. Finally, we found that a single model, trained with uPIT, can handle both two-speaker, and three-speaker speech mixtures.

1 Introduction

Having a conversation in a complex acoustic environment, with multiple noise sources and competing background speakers, is a task humans are remarkably good at [1, 2]. The problem that humans solve when they focus their auditory attention towards one audio signal in a complex mixture of signals is commonly known as the cocktail party problem [1, 2]. Despite intense research for more than half a century, a general machine based solution to the cocktail party problem is yet to be discovered [1–4]. A machine solution to the cocktail party problem is highly desirable for a vast range of applications. These include automatic meeting transcription, automatic captioning for audio/video recordings (e.g. YouTube), multi-party human-machine interaction (e.g. in the world of Internet of things (IoT)), and advanced hearing aids, where overlapping speech is commonly encountered.

Since the cocktail party problem was initially formalized [3], a large number of potential solutions have been proposed [5], and the most popular techniques originate from the field of Computational Auditory Scene Analysis (CASA) [6–10]. In CASA, different segmentation and grouping rules are used to group Time-Frequency (T-F) units that are believed to belong to the same speaker. The rules are typically hand-engineered and based on

heuristics such as pitch trajectory, common onset/offset, periodicity, etc. The grouped T-F units are then used to extract a particular speaker from the mixture signal. Another popular technique for multi-talker speech separation is Non-negative Matrix Factorization (NMF) [11–14]. The NMF technique uses non-negative dictionaries to decompose the spectrogram of the mixture signal into speaker specific activations, and from these activations an isolated target signal can be approximated using the dictionaries. For multi-talker speech separation, both CASA and NMF have led to limited success [4, 5] and the most successful techniques, before the deep learning era, are based on probabilistic models [15–17], such as factorial GMM-HMM [18], that model the temporal dynamics and the complex interactions of the target and competing speech signals. Unfortunately, these models assume and only work under closed-set speaker conditions, i.e. the identity of the speakers must be known *a priori*.

More recently, a large number of techniques based on deep learning [19] have been proposed, especially for Automatic Speech Recognition (ASR) [20–25], and speech enhancement [26–34]. Deep learning has also been applied in the context of multi-talker speech separation (e.g. [30]), although successful work has, similarly to NMF and CASA, mainly been reported for closed-set speaker conditions.

The limited success in deep learning based speaker independent multi-talker speech separation is partly due to the label permutation problem (which will be described in detail in Sec. 4). To the authors knowledge only four deep learning based works [35–38] exist, that have tried to address and solve the harder speaker independent multi-talker speech separation task.

In Weng *et al.* [35], which proposed the best performing system in the 2006 monaural speech separation and recognition challenge [4], the instantaneous energy was used to determine the training label assignment, which alleviated the label permutation problem and allowed separation of unknown speakers. Although this approach works well for two-speaker mixtures, it is hard to scale up to mixtures of three or more speakers.

Hershey *et al.* [36] have made significant progress with their Deep Clustering (DPCL) technique. In their work, a deep Recurrent Neural Network (RNN) is used to project the speech mixture into an embedding space, where T-F units belonging to the same speaker form a cluster. In this embedding space a clustering algorithm (e.g. K-means) is used to identify the clusters. Finally, T-F units belonging to the same clusters are grouped together and a binary mask is constructed and used to separate the speakers from the mixture signal. To further improve the model [39], another RNN is stacked on top of the first DPCL RNN to estimate continuous masks for each target speaker. Although DPCL show good performance, the technique is potentially limited because the objective function is based on the affinity between the sources in the embedding space, instead of the separated signals them-

selves. That is, low proximity in the embedding space does not necessarily imply perfect separation of the sources in the signal space.

Chen *et al.* [37, 40] proposed a related technique called Deep Attractor Network (DANet). Following DPCL, the DANet approach also learns a high-dimensional embedding of the mixture signals. Different from DPCL, however, it creates attractor points (cluster centers) in the embedding space, which attract the T-F units corresponding to each target speaker. The training is conducted in a way similar to the Expectation Maximization (EM) principle. The main disadvantage of DANet over DPCL is the added complexity associated with estimating attractor points during inference.

Recently, we proposed the Permutation Invariant Training (PIT) technique[1] [38] for attacking the speaker independent multi-talker speech separation problem and showed that PIT effectively solves the label permutation problem. However, although PIT solves the label permutation problem at training time, PIT does not effectively solve the permutation problem during inference, where the permutation of the separated signals at the frame-level is unknown. We denote the challenge of identifying this frame-level permutation, as the *speaker tracing problem*.

In this paper, we extend PIT and propose an utterance-level Permutation Invariant Training (uPIT) technique, which is a practically applicable, end-to-end, deep learning based solution for speaker independent multi-talker speech separation. Specifically, uPIT extends the frame-level PIT technique [38] with an utterance-level training criterion that effectively eliminates the need for additional speaker tracing or very large input/output contexts, which is otherwise required by the original PIT [38]. We achieve this using deep Long Short-Term Memory (LSTM) RNNs [41] that, during training, minimize the utterance-level separation error, hence forcing separated frames belonging to the same speaker to be aligned to the same output stream. This is unlike other techniques, such as DPCL and DANet, that require a distinct clustering step to separate speakers during inference. Furthermore, the computational cost associated with the uPIT training criterion is negligible compared to the computations required by the RNN during training and is zero during inference. We evaluated uPIT on the WSJ0 and Danish two- and three-talker mixed-speech separation tasks and found that uPIT outperforms techniques based on NMF and CASA, and compares favorably with DPCL and DANet. Furthermore, we show that models trained with uPIT generalize well to unseen speakers and languages, and finally, we found that a single model trained with uPIT can separate both two-speaker, and three-speaker speech mixtures.

[1] In [36], a related permutation free technique, which is similar to PIT for exactly two-speakers, was evaluated with negative results and conclusion.

The rest of the paper is organized as follows. In Sec. 2 we describe the monaural speech separation problem. In Sec. 3 we extend popular optimization criteria used in separating single-talker speech from noises, to multi-talker speech separation tasks. In Sec. 4 we discuss the label permutation problem and present the PIT framework. In Sec. 5 we introduce uPIT and show how an utterance-level permutation criterion can be combined with PIT. We report series of experimental results in Sec. 6 and conclude the paper in Sec. 7.

2 Monaural Speech Separation

The goal of monaural speech separation is to estimate the individual source signals $x_s[n],\ s = 1, 2, \cdots, S$ in a linearly mixed single-microphone signal

$$y[n] = \sum_{s=1}^{S} x_s[n], \tag{D.1}$$

based on the observed signal $y[n]$ only. In real situations, the received signals may be reverberated, i.e., the underlying clean signals are filtered before being observed in the mixture. In this condition, we aim at recovering the reverberated source signals $x_s[n]$, i.e., we are not targeting the dereverberated signals.

The separation is usually carried out in the T-F domain, in which the task can be cast as recovering the Short-Time discrete Fourier Transformation (STFT) of the source signals $X_s(t, f)$ for each time frame t and frequency bin f, given the mixed speech

$$Y(t, f) = \sum_{n=0}^{N-1} y[n + tL]w[n] \exp(-j2\pi nf/N), \tag{D.2}$$

where $w[n]$ is the analysis window of length N, the signal is shifted by an amount of L samples for each time frame $t = 0, 1, \cdots, T-1$, and each frequency bin $f = 0, 1, \cdots, N-1$ is corresponding to a frequency of $(f/N)f_s$ [Hz] when the sampling rate is f_s [Hz].

From the estimated STFT $\hat{X}_s(t, f)$ of each source signal, an inverse Discrete Fourier Transform (DFT)

$$\hat{x}_{s,t}[n] = \frac{1}{N} \sum_{f=0}^{N-1} \hat{X}_s(t, f) \exp(j2\pi nf/N) \tag{D.3}$$

can be used to construct estimated time-domain frames, and the overlap-add operation

$$\hat{x}_s[n] = \sum_{t=0}^{T-1} v[n - tL]\hat{x}_{s,t}[n - tL] \tag{D.4}$$

can be used to reconstruct the estimate $\hat{x}_s[n]$ of the original signal, where $v[n]$ is the synthesis window.

In a typical setup, however, only the STFT magnitude spectrum $A_s(t,f) \triangleq |X_s(t,f)|$ is estimated from the mixture during the separation process, and the phase of the mixed speech is used directly, when recovering the time domain waveforms of the separated sources. This is because phase estimation is still an open problem in the speech separation setup [42, 43]. Obviously, given only the magnitude of the mixed spectrum, $R(t,f) \triangleq |Y(t,f)|$, the problem of recovering $A_s(t,f)$ is under-determined, as there are an infinite number of possible $A_s(t,f)$, $s = 1,\ldots,S$ combinations that lead to the same $R(t,f)$. To overcome this problem, a supervised learning system has to learn from some training set $\mathbb{S}$ that contains corresponding observations of $R(t,f)$ and $A_s(t,f)$, $s = 1,\ldots,S$.

Let $\mathbf{a}_{s,i} = \left[A_s(i,1),\ A_s(i,2),\ \ldots\ A_s(i,\frac{N}{2}+1)\right]^T \in \mathbb{R}^{\frac{N}{2}+1}$ denote the single-sided magnitude spectrum for source s at frame i. Furthermore, let $\mathbf{A}_s \in \mathbb{R}^{(\frac{N}{2}+1)\times T}$ be the single-sided magnitude spectrogram for source s and all frames $i = 1,\ldots,T$, defined as $\mathbf{A}_s = [\mathbf{a}_{s,1},\ \mathbf{a}_{s,2},\ \ldots\ ,\mathbf{a}_{s,T}]$. Similarly, let $\mathbf{r}_i = \left[R(i,1),\ R(i,2),\ \ldots\ R(i,\frac{N}{2}+1)\right]^T$ be the single-sided magnitude spectrum of the observed signal at frame i and let $\mathbf{R} = [\mathbf{r}_1,\ \mathbf{r}_2,\ \ldots\ ,\mathbf{r}_T] \in \mathbb{R}^{(\frac{N}{2}+1)\times T}$ be the single-sided magnitude spectrogram for all frames $i = 1,\ldots,T$.

Furthermore, let us denote a supervector $\mathbf{z}_i = \left[\mathbf{a}_{1,i}^T, \mathbf{a}_{2,i}^T, \ldots, \mathbf{a}_{S,i}^T\right]^T \in \mathbb{R}^{S(\frac{N}{2}+1)}$, consisting of the stacked source magnitude spectra for each source $s = 1,\ldots,S$ at frame i and let $\mathbf{Z} = [\mathbf{z}_1,\ \mathbf{z}_2,\ \ldots\ ,\mathbf{z}_T] \in \mathbb{R}^{S(\frac{N}{2}+1)\times T}$ denote the matrix of all T supervectors. Finally, let $\mathbf{y}_i = \left[Y(i,1),\ Y(i,2),\ \ldots\ Y(i,\frac{N}{2}+1)\right]^T \in \mathbb{C}^{\frac{N}{2}+1}$ be the single-sided STFT of the observed mixture signal at frame i and $\mathbf{Y} = [\mathbf{y}_1,\ \mathbf{y}_2,\ \ldots\ ,\mathbf{y}_T] \in \mathbb{C}^{(\frac{N}{2}+1)\times T}$ be the STFT of the mixture signal for all T frames.

Our objective is then to train a deep learning model $g(\cdot)$, parameterized by a parameter set $\mathbf{\Phi}$, such that $g\left(d\left(\mathbf{Y}\right);\mathbf{\Phi}\right) = \mathbf{Z}$, where $d(\mathbf{Y})$ is some feature representation of the mixture signal: In a particularly simple situation, $d(\mathbf{Y}) = \mathbf{R}$, i.e., the feature representation is simply the magnitude spectrum of the observed mixture signal.

It is possible to directly estimate the magnitude spectra $\mathbf{Z}$ of all sources using a deep learning model. However, it is well-known (e.g. [27, 43]), that better results can be achieved if, instead of estimating $\mathbf{Z}$ directly, we first estimate a set of masks $M_s(t,f)$, $s = 1,\ldots,S$.

Let $\mathbf{m}_{s,i} = \left[M_s(i,1)\ ,\ M_s(i,2)\ ,\ \ldots\ M_s(i,\frac{N}{2}+1)\right]^T \in \mathbb{R}^{\frac{N}{2}+1}$ be the ideal mask (to be defined in detail in Sec. 3) for speaker s at frame i, and let $\mathbf{M}_s = [\mathbf{m}_{s,1},\ \mathbf{m}_{s,2},\ \ldots\ ,\mathbf{m}_{s,T}] \in \mathbb{R}^{(\frac{N}{2}+1)\times T}$ be the ideal mask for all T frames,

such that $\mathbf{A}_s = \mathbf{M}_s \circ \mathbf{R}$, where $\circ$ is the Hadamard product, i.e. element-wise product of two operands. Furthermore, let us introduce the mask supervector $\mathbf{u}_i = \left[\mathbf{m}_{1,i}^T\ \mathbf{m}_{2,i}^T\ \dots\ \mathbf{m}_{S,i}^T\right]^T \in \mathbb{R}^{S\left(\frac{N}{2}+1\right)}$ and the corresponding mask matrix $\mathbf{U} = [\mathbf{u}_1,\ \mathbf{u}_2,\ \dots\ ,\mathbf{u}_T] \in \mathbb{R}^{S\left(\frac{N}{2}+1\right)\times T}$. Our goal is then to find an estimate $\hat{\mathbf{U}}$ of $\mathbf{U}$, using a deep learning model, $h\left(\mathbf{R};\mathbf{\Phi}\right) = \hat{\mathbf{U}}$. Since, $\hat{\mathbf{U}} = [\hat{\mathbf{u}}_1,\ \hat{\mathbf{u}}_2,\ \dots\ ,\hat{\mathbf{u}}_T]$ and $\hat{\mathbf{u}}_i = \left[\hat{\mathbf{m}}_{1,i}^T\ \hat{\mathbf{m}}_{2,i}^T\ \dots\ \hat{\mathbf{m}}_{S,i}^T\right]^T$, the model output is easily divided into output streams corresponding to the estimated masks for each speaker $\hat{\mathbf{m}}_{s,i}$, and their resulting magnitudes are estimated as $\hat{\mathbf{a}}_{s,i} = \hat{\mathbf{m}}_{s,i} \circ \mathbf{r}_i$. The estimated time-domain signal for speaker s is then computed as the inverse DFT of $\hat{\mathbf{a}}_{s,i}$ using the phase of the mixture signal $\mathbf{y}_i$.

3 Masks and Training Criteria

Since masks are to be estimated as an intermediate step towards estimating magnitude spectra of source signals, we extend in the following three popular masks defined for separating single-talker speech from noises to the multi-talker speech separation task at hand.

3.1 Ideal Ratio Mask

The Ideal Ratio Mask (IRM) [27] for each source is defined as

$$M_s^{irm}(t,f) = \frac{|X_s(t,f)|}{\sum_{s=1}^{S}|X_s(t,f)|}. \tag{D.5}$$

When the phase of $\mathbf{Y}$ is used for reconstruction, the IRM achieves the highest Signal to Distortion Ratio (SDR) [44], when all sources have the same phase, (which is an invalid assumption in general). IRMs are constrained to $0 \leq M_s^{irm}(t,f) \leq 1$ and $\sum_{s=1}^{S} M_s^{irm}(t,f) = 1$ for all T-F units. This constraint can easily be satisfied using the softmax activation function.

Since $\mathbf{Y}$ is the only observed signal in practice and $\sum_{s=1}^{S}|X_s(t,f)|$ is unknown during separation, the IRM is not a desirable target for the problem at hand. Nevertheless, we report IRM results as an upper performance bound since the IRM is a commonly used training target for deep learning based monaural speech separation [31, 32].

3.2 Ideal Amplitude Mask

Another applicable mask is the Ideal Amplitude Mask (IAM) (known as FFT-mask in [27]), or simply Amplitude Mask (AM), when estimated by a deep

learning model. The IAM is defined as

$$M_s^{iam}(t,f) = \frac{|X_s(t,f)|}{|Y(t,f)|}. \tag{D.6}$$

Through IAMs we can construct the exact $|X_s(t,f)|$ given the magnitude spectra of the mixed speech $|Y(t,f)|$. If the phase of each source equals the phase of the mixed speech, the IAM achieves the highest SDR. Unfortunately, as with the IRM, this assumption is not satisfied in most cases. IAMs satisfy the constraint that $0 \leq M_s^{iam}(t,f) \leq \infty$, although we found empirically that the majority of the T-F units are in the range of $0 \leq M_s^{iam}(t,f) \leq 1$. For this reason, softmax, sigmoid and ReLU are all possible output activation functions for estimating IAMs.

3.3 Ideal Phase Sensitive Mask

Both IRM and IAM do not consider phase differences between source signals and the mixture. This leads to sub-optimal results, when the phase of the mixture is used for reconstruction. The Ideal Phase Sensitive Mask (IPSM) [43, 45]

$$M_s^{ipsm}(t,f) = \frac{|X_s(t,f)|\cos(\theta_y(t,f) - \theta_s(t,f))}{|Y(t,f)|}, \tag{D.7}$$

however, takes phase differences into consideration, where θ_y and θ_s are the phases of mixed speech $Y(t,f)$ and source $X_s(t,f)$, respectively. Due to the phase-correcting term, the IPSM sums to one, i.e. $\sum_{s=1}^{S} M_s^{ipsm}(t,f) = 1$. Note that since $|\cos(\cdot)| \leq 1$ the IPSM is smaller than the IAM, especially when the phase difference between the mixed speech and the source is large.

Even-though the IPSM in theory is unbounded, we found empirically that the majority of the IPSM is in the range of $0 \leq M_s^{ipsm}(t,f) \leq 1$. Actually, in our study we have found that approximately 20% of IPSMs are negative. However, those negative IPSMs usually are very close to zero. To account for this observation, we propose the Ideal Non-negative Phase Sensitive Mask (INPSM), which is defined as

$$M_s^{inpsm}(t,f) = max(0, M_s^{ipsm}(t,f)). \tag{D.8}$$

For estimating the IPSM and INPSM, Softmax, Sigmoid, tanh, and ReLU are all possible activation functions, and similarly to the IAM, when the IPSM is estimated by a deep learning model we refer to it as Phase Sensitive Mask (PSM).

3.4 Training Criterion

Since we first estimate masks, through which the magnitude spectrum of each source can be estimated, the model parameters can be optimized to minimize

the Mean Squared Error (MSE) between the estimated mask $\hat{M}_s$ and one of the target masks defined above as

$$J_m = \frac{1}{B}\sum_{s=1}^{S} \|\hat{\mathbf{M}}_s - \mathbf{M}_s\|_F^2, \tag{D.9}$$

where $B = T \times N \times S$ is the total number of T-F units over all sources and $\|\cdot\|_F$ is the Frobenius norm. This approach comes with two problems. First, in silence segments, $|X_s(t,f)| = 0$ and $|Y(t,f)| = 0$, so that the target masks $M_s(t,f)$ are not well defined. Second, what we really care about is the error between the reconstructed source signal and the true source signal.

To overcome these limitations, recent works [27] directly minimize the MSE

$$\begin{aligned} J_a &= \frac{1}{B}\sum_{s=1}^{S} \|\hat{\mathbf{A}}_s - \mathbf{A}_s\|_F^2 \\ &= \frac{1}{B}\sum_{s=1}^{S} \|\hat{\mathbf{M}}_s \circ \mathbf{R} - \mathbf{A}_s\|_F^2 \end{aligned} \tag{D.10}$$

between the estimated magnitude, i.e. $\hat{\mathbf{A}}_s = \hat{\mathbf{M}}_s \circ \mathbf{R}$ and the true magnitude $\mathbf{A}_s$. Note that in silence segments $A_s(t,f) = 0$ and $R(t,f) = 0$, so the accuracy of mask estimation does not affect the training criterion for those segments. Furthermore, using Eq. (D.10) the IAM is estimated as an intermediate step.

When the PSM is used, the cost function becomes

$$J_{psm} = \frac{1}{B}\sum_{s=1}^{S} \|\hat{\mathbf{M}}_s \circ \mathbf{R} - \mathbf{A}_s \circ \cos(\boldsymbol{\theta}_y - \boldsymbol{\theta}_s)\|_F^2. \tag{D.11}$$

In other words, using PSMs is as easy as replacing the original training targets with the phase discounted targets. Furthermore, when Eq. (D.11) is used as a cost function, the IPSM is the upper bound achievable on the task [43].

4 Permutation Invariant Training

4.1 Conventional Multi-Talker Separation

A natural, and commonly used, approach for deep learning based speech separation is to cast the problem as a multi-class [30, 35, 46] regression problem as depicted in Fig. D.1.

For this conventional two-talker separation model, J frames of feature vectors of the mixed signal $\mathbf{Y}$ are used as the input to some deep learning model

e.g. a feed-forward Deep Neural Network (DNN), Convolutional Neural Network (CNN), or LSTM RNN, to generate M frames of masks for each talker. Specifically, if $M = 1$, the output of the model can be described by the vector $\hat{\mathbf{u}}_i = \left[\hat{\mathbf{m}}_{1,i}^T \; \hat{\mathbf{m}}_{2,i}^T\right]^T$ and the sources are separated as $\hat{\mathbf{a}}_{1,i} = \hat{\mathbf{m}}_{1,i} \circ \mathbf{r}_i$ and $\hat{\mathbf{a}}_{2,i} = \hat{\mathbf{m}}_{2,i} \circ \mathbf{r}_i$, for sources $s = 1, 2$, respectively.

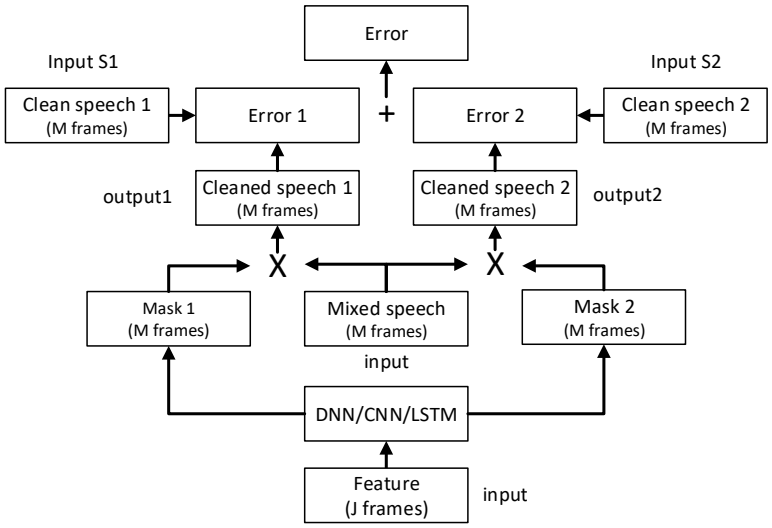


Fig. D.1: The conventional two-talker speech separation model.

4.2 The Label Permutation Problem

During training, the error (e.g. using Eq. (D.11)) between the clean magnitude spectra $\mathbf{a}_{1,i}$ and $\mathbf{a}_{2,i}$ and their estimated counterparts $\hat{\mathbf{a}}_{1,i}$ and $\hat{\mathbf{a}}_{2,i}$ needs to be computed. However, since the model estimates the masks $\hat{\mathbf{m}}_{1,i}$ and $\hat{\mathbf{m}}_{2,i}$ simultaneously, and they depend on the same input mixture, it is unknown in advance whether the resulting output vector $\hat{\mathbf{u}}_i$ is ordered as $\hat{\mathbf{u}}_i = \left[\hat{\mathbf{m}}_{1,i}^T \; \hat{\mathbf{m}}_{2,i}^T\right]^T$ or $\hat{\mathbf{u}}_i = \left[\hat{\mathbf{m}}_{2,i}^T \; \hat{\mathbf{m}}_{1,i}^T\right]^T$. That is, the permutation of the output masks is unknown.

A naïve approach to train a deep learning separation model, without exact knowledge about the permutation of the output masks, is to use a constant permutation as illustrated by Fig. D.1. Although such a training approach works for simple cases e.g. female speakers mixed with male speakers, in which case *a priori* convention can be made that e.g. the first output stream contains the female speaker, while the second output stream is paired with

the male speaker, the training fails if the training set consists of many utterances spoken by many speakers of both genders.

This problem is referred to as the label permutation (or ambiguity) problem in [35, 36]. Due to this problem, prior arts perform poorly on speaker independent multi-talker speech separation.

4.3 Permutation Invariant Training

Our solution to the label permutation problem is illustrated in Fig. D.2 and is referred to as Permutation Invariant Training (PIT) [38].

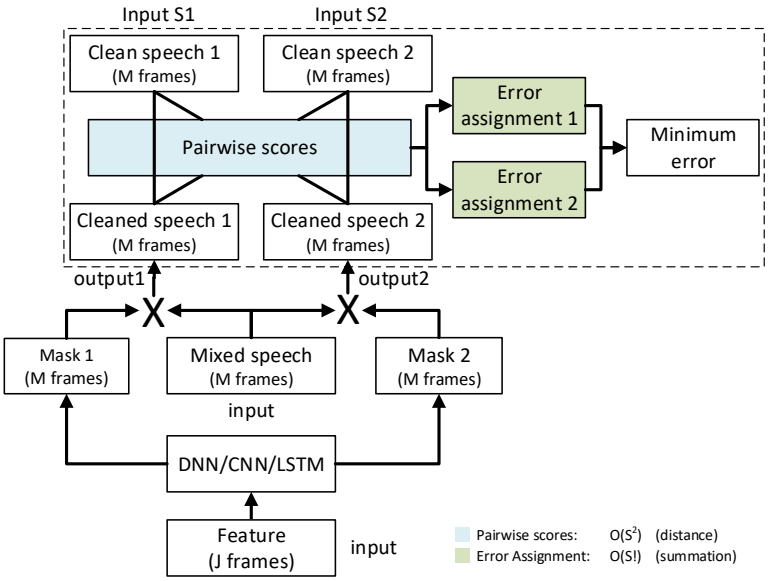


Fig. D.2: The two-talker speech separation model with permutation invariant training.

In the model depicted in Fig. D.2 (and unlike the conventional model in Fig. D.1) the reference signals are given as *a set* instead of an ordered list. In other words, the same training result is obtained, no matter in which order these references are listed. This behavior is achieved with PIT highlighted inside the dashed rectangle in Fig. D.2. Specifically, following the notation from Sec. 2, we associate the reference signals for speaker one and two, i.e. $\mathbf{a}_{1,i}$ and $\mathbf{a}_{2,i}$, to the output masks $\hat{\mathbf{m}}_{1,i}$ and $\hat{\mathbf{m}}_{2,i}$, by computing the (total of S^2) pairwise MSEs between each reference signal $\mathbf{a}_{s,i}$ and each estimated source $\hat{\mathbf{a}}_{s,i}$. We then determine the (total of $S!$) possible permutations between the references and the estimated sources, and compute the *per-permutation-loss* for each permutation. That is, for the two-speaker case in Fig. D.2 we compute

the *per-permutation-loss* for the two candidate output vectors $\hat{\mathbf{u}}_i = \left[\hat{\mathbf{m}}_{1,i}^T \; \hat{\mathbf{m}}_{2,i}^T\right]^T$ and $\hat{\mathbf{u}}_i = \left[\hat{\mathbf{m}}_{2,i}^T \; \hat{\mathbf{m}}_{1,i}^T\right]^T$. The permutation with the lowest MSE is chosen and the model is optimized to reduce this least MSE. In other words, we simultaneously conduct label assignment and error evaluation. Similarly to prior arts, we can use J, and M successive input, and output frames, respectively, (i.e., a *meta-frame*) to exploit the contextual information. Note that only S^2 pairwise MSEs are required (and not $S!$) to compute the *per-permutation-loss* for all $S!$ possible permutations. Since $S!$ grows much faster than S^2, with respect to S, and the computational complexity of the pairwise MSE is much larger than the *per-permutation-loss* (sum of pairwise MSEs), PIT can be used with a large number of speakers, i.e. $S \gg 2$.

During inference, the only information available is the mixed speech, but speech separation can be directly carried out for each input meta-frame, for which an output meta-frame with M frames of speech is estimated. Due to the PIT training criterion, the permutation will stay the same for frames inside the same output meta-frame, but may change across output meta-frames. In the simplest setup, we can just assume that permutations do not change across output meta-frames, when reconstructing the target speakers. However, this usually leads to unsatisfactory results as reported in [38]. To achieve better performance, speaker tracing algorithms, that identify the permutations of output meta-frames with respect to the speakers, need to be developed and integrated into the PIT framework or applied on top of the output of the network.

5 Utterance-Level PIT

Several ways exist for identifying the permutation of the output meta-frames, i.e. solving the tracing problem. For example, in CASA a related problem referred to as the Sequential Organization Problem has been addressed using a model-based sequential grouping algorithm [9]. Although moderately successful for co-channel speech separation, where prior knowledge about the speakers is available, this method is not easily extended to the speaker independent case with multiple speakers. Furthermore, it is not easily integrated into a deep learning framework.

A more straight-forward approach might be to determine a change in permutation by comparing MSEs for different permutations of output masks measured on the overlapping frames of adjacent output meta-frames. However, this approach has two major problems. First, it requires a separate tracing step, which may complicate the model. Second, since the permutation of later frames depends on that of earlier frames, one incorrect assignment at an earlier frame would completely switch the permutation for all frames after it,

even if the assignment decisions for the remaining frames are all correct.

In this work we propose utterance-level Permutation Invariant Training (uPIT), a simpler yet more effective approach to solve the tracing problem and the label permutation problem than original PIT. Specifically, we extend the frame-level PIT technique with the following utterance-level cost function:

$$J_{\phi^*} = \frac{1}{B}\sum_{s=1}^{S} \|\hat{\mathbf{M}}_s \circ \mathbf{R} - \mathbf{A}_{\phi^*(s)} \circ \cos(\boldsymbol{\theta}_y - \boldsymbol{\theta}_{\phi^*(s)})\|_F^2, \tag{D.12}$$

where ϕ^* is the permutation that minimizes the utterance-level separation error defined as

$$\phi^* = \underset{\phi\in\mathcal{P}}{\operatorname{argmin}} \sum_{s=1}^{S} \|\hat{\mathbf{M}}_s \circ \mathbf{R} - \mathbf{A}_{\phi(s)} \circ \cos(\boldsymbol{\theta}_y - \boldsymbol{\theta}_{\phi(s)})\|_F^2, \tag{D.13}$$

and $\mathcal{P}$ is the symmetric group of degree S, i.e. the set of all $S!$ permutations.

In original PIT, the optimal permutation (in MSE sense) is computed and applied *for each* output meta-frame. This implies that consecutive meta-frames might be associated with different permutations, and although PIT solves the label permutation problem, it does not solve the speaker tracing problem. With uPIT, however, the permutation corresponding to the minimum utterance-level separation error is used *for all* frames in the utterance. In other words, the pair-wise scores in Fig. D.2 are computed for the whole utterance assuming all output frames follow the same permutation. Using the same permutation *for all* frames in the utterance might imply that a non-MSE-optimal permutation is used for individual frames within the utterance. However, the intuition behind uPIT is that since the permutation resulting in the minimum utterance-level separation error is used, the number of non-optimal permutations is small and the model sees enough correctly permuted frames to learn an efficient separation model. For example, the output vector $\hat{\mathbf{u}}_i$ of a perfectly trained two-talker speech separation model, given an input utterance, should ideally be $\hat{\mathbf{u}}_i = \left[\hat{\mathbf{m}}_{1,i}^T\ \hat{\mathbf{m}}_{2,i}^T\right]^T$, or $\hat{\mathbf{u}}_i = \left[\hat{\mathbf{m}}_{2,i}^T\ \hat{\mathbf{m}}_{1,i}^T\right]^T\ \forall\ i = 1,\ldots,T$, i.e. the output masks should follow the same permutation for all T frames in the utterance. Fortunately, using Eq. (D.12) as a training criterion, for deep learning based speech separation models, this seems to be the case in practice (See Sec. 6 for examples).

Since utterances have variable length, and effective separation presumably requires exploitation of long-range signal dependencies, models such as DNNs and CNNs are no longer good fits. Instead, we use deep LSTM RNNs and Bi-directional Long Short-Term Memory (BLSTM) RNNs together with uPIT to learn the masks. Different from PIT, in which the input layer and each output layer has $N \times T$ and $N \times M$ units, respectively, in uPIT,

both input and output layers have N units (adding contextual frames in the input does not help for LSTMs). With deep LSTMs, the utterance is evaluated frame-by-frame exploiting the whole past history information at each layer. When BLSTMs are used, the information from the past and future (i.e., across the whole utterance) is stacked at each layer and used as the input to the subsequent layer. With uPIT, during inference we don't need to compute pairwise MSEs and errors of each possible permutation and no additional speaker tracing step is needed. We simply assume a constant permutation and treat the same output mask to be from the same speaker for all frames. This makes uPIT a simple and attractive solution.

6 Experimental Results

We evaluated uPIT on various setups and all models were implemented using the Microsoft Cognitive Toolkit (CNTK) [47, 48][2]. The models were evaluated on their potential to improve the Signal-to-Distortion Ratio (SDR) [44] and the Perceptual Evaluation of Speech Quality (PESQ) [49] score, both of which are metrics widely used to evaluate speech enhancement performance for multi-talker speech separation tasks.

6.1 Datasets

We evaluated uPIT on the WSJ0-2mix, WSj0-3mix[3] and Danish-2mix datasets using 129-dimensional STFT magnitude spectra computed with a sampling frequency of 8 kHz, a frame size of 32 ms and a 16 ms frame shift.

The WSJ0-2mix dataset was introduced in [36] and was derived from the WSJ0 corpus [50]. The 30h training set and the 10h validation set contain two-speaker mixtures generated by randomly selecting from 49 male and 51 female speakers and utterances from the WSJ0 training set si_tr_s, and mixing them at various Signal-to-Noise Ratios (SNRs) uniformly chosen between 0 dB and 5 dB. The 5h test set was similarly generated using utterances from 16 speakers from the WSJ0 validation set si_dt_05 and evaluation set si_et_05. The WSJ0-3mix dataset was generated using a similar approach but contains mixtures of speech from three talkers.

The Danish-2mix dataset is based on a corpus[4] with approximately 560 speakers each speaking 312 utterances with average utterance duration of approximately 5 sec. The dataset was constructed by randomly selecting a set of 45 male and 45 female speakers from the corpus, and then allocating 232

[2] Available at: `https://www.cntk.ai/`

[3] Available at: `http://www.merl.com/demos/deep-clustering`

[4] Available at: `http://www.nb.no/sbfil/dok/nst_taledat_dk.pdf`

and 40 utterances from each speaker to generate mixed speech in the training, and validation set, respectively. A number of 40 utterances from each of another 45 male and 45 female speakers were randomly selected to construct the Open-Condition (OC) (unseen speaker) test set. Speech mixtures were constructed similarly to the WSJ0-2mix with SNRs selected uniformly between 0 dB and 5 dB. Similarly to the WSJ0-2mix dataset we constructed 20k and 5k mixtures in total in the training and validation set, respectively, and 3k mixtures for the OC test set.

In our study, the validation set is used to find initial hyper-parameters and to evaluate Closed-Condition (CC) (seen speaker) performance, similarly to [36, 38, 39].

6.2 Permutation Invariant Training

We first evaluated the original frame-level PIT on the two-talker separation dataset WSJ0-2mix, and differently from [38], we fixed the input dimension to 51 frames, to isolate the effect of a varying output dimension. In PIT, the input window and output window sizes are fixed. For this reason, we can use DNNs and CNNs. The DNN model has three hidden layers each with 1024 ReLU units. In (inChannel, outChannel)-(strideW, strideH) format, the CNN model has one $(1, 64) - (2, 2)$, four $(64, 64) - (1, 1)$, one $(64, 128) - (2, 2)$, two $(128, 128) - (1, 1)$, one $(128, 256) - (2, 2)$, and two $(256, 256) - (1, 1)$ convolution layers with 3×3 kernels, a 7×17 average pooling layer and a 1024-unit ReLU layer. The input to the models is the stack (over multiple frames) of the 129-dimensional STFT spectral magnitude of the speech mixture. The output layer $\hat{\mathbf{u}}_i$ is divided into S output masks/streams for S-talker mixed speech as $\hat{\mathbf{u}}_i = [\hat{\mathbf{m}}_{1,i}\ ;\ \hat{\mathbf{m}}_{2,i}\ ;\ \ldots\ ;\ \hat{\mathbf{m}}_{S,i}]^T$. Each output mask vector $\hat{\mathbf{m}}_{s,i}$ has a dimension of $129 \times M$, where M is the number of frames in the output meta-frame.

In Fig. D.3 we present the DNN training progress as measured by the MSE on the training and validation set with conventional training (CONV-DNN) and PIT on the WSJ0-2mix datasets described in subsection 6.1. We also included the training progress for another conventionally trained model but with a slightly modified version of the WSJ0-2mix dataset, where speaker labels have been randomized (CONV-DNN-RAND).

The WSJ0-2mix dataset, used in [36], was designed such that speaker one was always assigned the most energy, and consequently speaker two the lowest, when scaling to a given SNR. Previous work [35] has shown that such speaker energy patterns are an effective discriminative feature, which is clearly seen in Fig. D.3, where the CONV-DNN model achieves considerably lower training and validation MSE than the CONV-DNN-RAND model, which hardly decreases in either training or validation MSE due to the label permutation problem [35, 36]. In contrast, training converges quickly to a very low MSE when PIT is used.

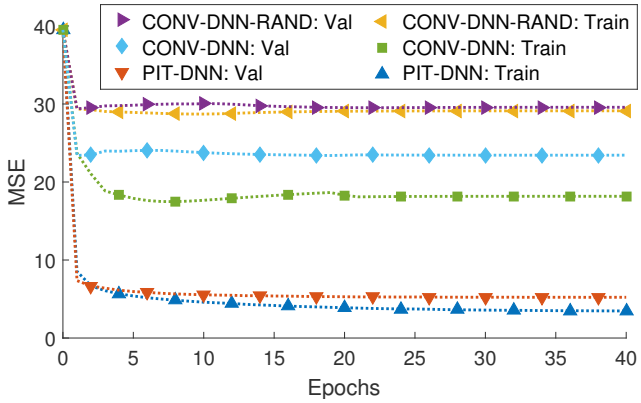


Fig. D.3: MSE over epochs on the WSJ0-2mix training and validation sets with conventional training and PIT.

Table D.1: SDR improvements (dB) for different separation methods on the WSJ0-2mix dataset using PIT.

Method	Input\Output window	Opt. Assign.		Def. Assign.	
		CC	OC	CC	OC
PIT-DNN	51\51	6.8	6.7	**5.2**	**5.2**
PIT-DNN	51\5	**10.3**	**10.2**	-0.8	-0.8
PIT-CNN	51\51	9.6	9.6	**7.6**	**7.5**
PIT-CNN	51\5	**10.9**	**11.0**	-1.0	-0.9
IRM	-	12.4	12.7	12.4	12.7
IPSM	-	14.9	15.1	14.9	15.1

In Table D.1 we summarize the SDR improvement in dB from different frame-level PIT separation configurations for two-talker mixed speech in closed condition (CC) and open condition (OC). In these experiments each frame was reconstructed by averaging over all output meta-frames that contain the same frame. In the default assignment (def. assign.) setup, a constant output mask permutation is assumed across frames (which is an invalid assumption in general). This is the maximum achievable SDR improvement using PIT without the utterance-level training criterion and without an additional tracing step. In the optimal assignment (opt. assign.) setup, the output-mask permutation for each output meta-frame is determined based on the true target, i.e. oracle information. This reflects the separation performance within each segment (meta-frame) and is the improvement achievable when the speakers are correctly separated. The gap between these two values indicates the possible contribution from speaker tracing. As a reference, we also provided the IRM and IPSM results.

From the table we can make several observations. First, PIT can already

Table D.2: SDR improvements (dB) for different separation methods on the WSJ0-2mix dataset using uPIT.

Method	Mask Type	Activation Function	Opt. Assign.		Def. Assign.	
			CC	OC	CC	OC
uPIT-BLSTM	AM	softmax	**10.4**	**10.3**	**9.0**	**8.7**
uPIT-BLSTM	AM	sigmoid	8.3	8.3	7.1	7.2
uPIT-BLSTM	AM	ReLU	9.9	9.9	8.7	8.6
uPIT-BLSTM	AM	Tanh	8.5	8.6	7.5	7.5
uPIT-BLSTM	PSM	softmax	10.3	10.2	9.1	9.0
uPIT-BLSTM	PSM	sigmoid	10.5	10.4	9.2	9.1
uPIT-BLSTM	PSM	ReLU	**10.9**	**10.8**	**9.4**	**9.4**
uPIT-BLSTM	PSM	Tanh	10.4	10.3	9.0	8.9
uPIT-BLSTM	NPSM	softmax	8.7	8.6	7.5	7.3
uPIT-BLSTM	NPSM	sigmoid	**10.6**	**10.6**	**9.4**	**9.3**
uPIT-BLSTM	NPSM	ReLU	8.8	8.8	7.6	7.6
uPIT-BLSTM	NPSM	Tanh	10.1	10.0	8.9	8.8
uPIT-LSTM	PSM	ReLU	**9.8**	**9.8**	7.0	**7.0**
uPIT-LSTM	PSM	sigmoid	**9.8**	9.6	**7.1**	6.9
uPIT-LSTM	NPSM	ReLU	**9.8**	**9.8**	**7.1**	**7.0**
uPIT-LSTM	NPSM	sigmoid	9.2	9.2	6.8	6.8
PIT-BLSTM	PSM	ReLU	**11.7**	**11.7**	**-1.7**	-1.9
PIT-BLSTM	PSM	sigmoid	**11.7**	**11.7**	**-1.7**	**-1.7**
PIT-BLSTM	NPSM	ReLU	**11.7**	**11.7**	**-1.7**	-1.8
PIT-BLSTM	NPSM	sigmoid	11.6	11.6	-1.6	**-1.7**
IRM	-	-	12.4	12.7	12.4	12.7
IPSM	-	-	14.9	15.1	14.9	15.1

achieve 7.5 dB SDR improvement (def. assign.), even though the model is very simple. Second, as we reduce the output window size, we can improve the separation performance within each window and achieve better SDR improvement, if speakers are correctly traced (opt. assign.). However, when output window size is reduced, the output mask permutation changes more frequently as indicated by the poor default assignment performance. Speaker tracing thus becomes more important given the larger gap between the optimal assignment and default assignment. Third, PIT generalizes well to unseen speakers, since the performances on the open and closed conditions are very close. Fourth, powerful models such as CNNs consistently outperform DNNs, but the gain diminishes when the output window size is small.

6.3 Utterance-Level Permutation Invariant Training

As indicated by Table D.1, an accurate output mask permutation is critical to further improve the separation quality. In this subsection we evaluate the uPIT technique as discussed in Sec. 5 and the results are summarized in Table D.2.

Due to the formulation of the uPIT cost function in Eq. (D.12) and Eq. (D.13), and to utilize long-range context, RNNs are the natural choice, and in this set of experiments, we used LSTM RNNs. All the uni-directional LSTMs (uPIT-LSTM) evaluated have 3 LSTM layers each with 1792 units and all the bi-directional LSTMs (uPIT-BLSTM) have 3 BLSTM layers each with 896 units, so that both models have similar number of parameters.

All models contain random dropouts when fed from a lower layer to a higher layer and were trained with a dropout rate of 0.5. Note that, since we used Nvidia's cuDNN implementation of LSTMs, to speed up training, we were unable to apply dropout across time steps, which was adopted by the best DPCL model [39] and is known to be more effective, both theoretically and empirically, than the simple dropout strategy used in this work [51].

In all the experiments reported in Table D.2 the maximum epoch is set to 200 although we noticed that further performance improvement is possible with additional training epochs. Note that the epoch size of 200 seems to be significantly larger than that in PIT as indicated in Fig. D.3. This is likely because in PIT each frame is used by T ($T = 51$) training samples (input meta-frames) while in uPIT each frame is used just once in each epoch.

The learning rates were set to 2×10^{-5} per sample initially and scaled down by 0.7 when the training objective function value increases on the training set. The training was terminated when the learning rate got below 10^{-10}. Each minibatch contains 8 randomly selected utterances.

As a related baseline, we also include PIT-BLSTM results in Table D.2. These models were also trained using LSTMs with whole utterances instead of meta-frames. The only difference between these models and uPIT models is that uPIT models use the utterance-level training criterion defined in Eqs. (D.12) and (D.13), instead of the meta-frame based criterion used by PIT.

6.3.1 uPIT Training Progress

In Fig. D.4 we present a representative example of the BLSTM training progress, as measured by the MSE of the two-talker mixed speech training and validation set, using Eq. (D.12). We see that the training and validation MSEs are both steadily decreasing as function of epochs, hence uPIT, similarly to PIT, effectively solves the label permutation problem.

6.3.2 uPIT Performance for Different Setups

From Table D.2, we can notice several things. First, with uPIT, we can significantly improve the SDR with default assignment over original PIT. In fact, a 9.4 dB SDR improvement on both CC and OC sets can be achieved by simply assuming a constant output mask permutation (def. assign.), which compares favorably to 7.6 dB (CC) and 7.5 dB (OC) achieved with deep CNNs

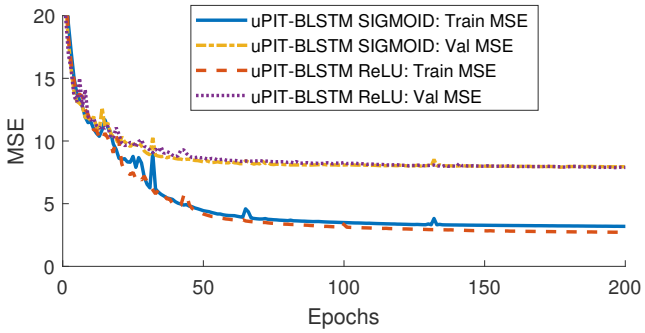


Fig. D.4: MSE over epochs on the WSJ0-2mix PSM training and validation sets wit uPIT.

combined with PIT. We want to emphasize that this is achieved through Eqs. (D.12) and (D.13), and not by using BLSTMs because the corresponding PIT-BLSTM default assignment results are so much worse, even though the optimal assignment results are the best among all models. The latter may be explained from the PIT objective function that attempts to obtain a constant output mask permutation at the meta-frame-level, which for small meta-frames is assumed easier compared to the uPIT objective function, that attempts to obtain a constant output mask permutation throughout the whole utterance. Second, we can achieve better SDR improvement over the AM using PSM and NPSM training criteria. This indicates that including phase information does improve performance, even-though it was used implicitly via the cosine term in Eq. (D.12). Third, with uPIT the gap between optimal assignment and default assignment is always less than 1.5 dB across different setups, hence additional improvements from speaker tracing algorithms is limited to 1.5 dB.

6.3.3 Two-Stage Models and Reduced Dropout Rate

It is well known that cascading DNNs can improve performance for certain deep learning based applications [39, 52–54]. In Table D.3 we show that a similar principle of cascading two BLSTM models into a two-stage model (-ST models in Table D.3) can lead to improved performance over the models presented in Table D.2. In Table D.3 we also show that improved performance, with respect to the same models, can be achieved with additional training epochs combined with a reduced dropout rate (-RD models in Table D.3). Specifically, if we continue the training of the two best performing models from Table D.2 (i.e. uPIT-BLSTM-PSM-ReLU and uPIT-BLSTM-NPSM-Sigmoid) with 200 additional training epochs at a reduced dropout rate of 0.3, we see an improvement of 0.1 dB. Even larger improvements can be achieved with the two-stage approach, where an estimated mask is com-

Table D.3: Further improvement on the WSJ0-2mix dataset with additional training epochs with reduced dropout (-RD) or stacked models (-ST)

Method	Mask Type	Activation Function	Opt. Assign.		Def. Assign.	
			CC	OC	CC	OC
uPIT-BLSTM-RD	PSM	ReLU	11.0	11.0	9.5	9.5
uPIT-BLSTM-ST	PSM	ReLU	**11.7**	**11.7**	10.0	**10.0**
uPIT-BLSTM-RD	NPSM	Sigmoid	10.7	10.7	9.5	9.4
uPIT-BLSTM-ST	NPSM	Sigmoid	11.5	11.5	**10.1**	**10.0**
IRM	-	-	12.4	12.7	12.4	12.7
IPSM	-	-	14.9	15.1	14.9	15.1

Table D.4: SDR (dB) improvements on test sets of WSJ0-2mix divided into same and opposite gender mixtures

Method	Config	CC		OC	
		Same	Opp.	Same	Opp.
uPIT-BLSTM-RD	PSM-ReLU	7.5	11.5	7.1	11.6
uPIT-BLSTM-ST	PSM-ReLU	7.8	**12.1**	**7.5**	**12.2**
uPIT-BLSTM-RD	NPSM-Sigmoid	7.5	11.5	7.0	11.5
uPIT-BLSTM-ST	NPSM-Sigmoid	**8.0**	**12.1**	**7.5**	12.1
IRM	-	12.2	12.7	12.4	12.9
IPSM	-	14.6	15.1	14.9	15.3

puted as the average mask from two BLSTM models as

$$\hat{\mathbf{M}}_s = \frac{\hat{\mathbf{M}}_s^{(1)} + \hat{\mathbf{M}}_s^{(2)}}{2}. \tag{D.14}$$

The mask $\hat{\mathbf{M}}_s^{(1)}$ is from an -RD model that serves as a first-stage model, and $\hat{\mathbf{M}}_s^{(2)}$ is the output mask from a second-stage model. The second-stage model is trained using the original input features as well as the mask $\hat{\mathbf{M}}_s^{(1)}$ from the first-stage model. The intuition behind this architecture is that the second-stage model will learn to correct the errors made by the first-stage model. Table D.3 shows that the two-stage models (-ST models) always outperform the single-stage models (-RD models) and overall, a 10 dB SDR improvement can be achieved on this task using a two-stage approach.

6.3.4 Opposite Gender vs. Same Gender.

Table D.4 reports SDR (dB) improvements on test sets of WSJ0-2mix divided into opposite-gender (Opp.) and same-gender (Same). From this table we can clearly see that our approach achieves much better SDR improvements on the opposite-gender mixed speech than the same-gender mixed speech, although

Table D.5: SDR (dB) and PESQ improvements on WSJ0-2mix and Danish-2mix with uPIT-BLSTM-PSM-ReLU trained on WSJ0-2mix and a combination of two languages.

Trained on	WSJ0-2mix		Danish-2mix	
	SDR	PESQ	SDR	PESQ
WSJ0-2mix	9.4	0.62	8.1	0.40
+Danish-2mix	8.8	0.58	10.6	0.51
IRM	12.7	2.11	15.2	1.90
IPSM	15.1	2.10	17.7	1.90

the gender information is not explicitly used in our model and training procedure. In fact, for the opposite-gender condition, the SDR improvement is already very close to the IRM result. These results agree with breakdowns from other works [36, 39] and generally indicate that same-gender mixed speech separation is a harder task.

6.3.5 Multi-Language Models

To further understand the properties of uPIT, we evaluated the uPIT-BLSTM-PSM-ReLU model trained on WSJ0-2mix (English) on the Danish-2mix test set. The results of this is reported in Table D.5. An interesting observation, is that although the system has never seen Danish speech, it performs remarkably well in terms of SDR, when compared to the IRM (oracle) values. These results indicate, that the separation ability learned with uPIT generalizes well, not only across speakers, but also across languages. In terms of PESQ, we see a somewhat larger performance gap with respect to the IRM. This might be explained by the fact that SDR is a waveform matching criteria and does not necessarily reflect perceived quality as well as PESQ. Furthermore, we note that the PESQ improvements are similar to what have been reported for DNN based speech enhancement systems [32].

We also trained a model with the combination of English and Danish datasets and evaluated the models on both languages. The results of these experiments are summarized in Table D.5. Table D.5, indicate that by including Danish data, we can achieve better performance on the Danish dataset, at the cost of slightly worse performance on the English dataset. Note that while doubling the training set, we did not change the model size. Had we done this, performance would likely improve on both languages.

6.3.6 Summary of Multiple 2-Speaker Separation Techniques

Table D.6 summarizes SDR (dB) and PESQ improvements for different separation methods on the WSJ0-2mix dataset. From the table we can observe that the models trained with PIT already achieve similar or better SDR than

Table D.6: SDR (dB) and PESQ improvements for different separation methods on the WSJ0-2mix dataset without additional tracing (i.e., def. assign.).

Method	Config	PESQ Imp.		SDR Imp.	
		CC	OC	CC	OC
Oracle NMF [36]	-	-	-	5.1	-
CASA [36]	-	-	-	2.9	3.1
DPCL [36]	-	-	-	5.9	5.8
DPCL+ [37]	-	-	-	-	9.1
DANet [37]	-	-	-	-	9.6
DANet‡ [37]	-	-	-	-	10.5
DPCL++ [39]	-	-	-	-	9.4
DPCL++‡ [39]	-	-	-	-	**10.8**
PIT-DNN	51\51	0.24	0.23	5.2	5.2
PIT-CNN	51\51	0.52	0.50	7.6	7.6
uPIT-BLSTM	PSM-ReLU	0.66	0.62	9.4	9.4
uPIT-BLSTM-ST	PSM-ReLU	**0.86**	**0.82**	**10.0**	10.0
IRM	-	2.15	2.11	12.4	12.7
IPSM	-	2.14	2.10	14.9	15.1

‡ indicates curriculum training.

the original DPCL [36], respectively, with DNNs and CNNs. Using the uPIT training criteria, we improve on PIT and achieve comparable performance with DPCL+, DPCL++ and DANet models[5] reported in [37, 39], which used curriculum training [55], and recurrent dropout [51]. Note that, both uPIT and PIT models are much simpler than DANet, DPCL, DPCL+, and DPCL++, because uPIT and PIT models do not require any clustering step during inference or estimation of attractor points, as required by DANet.

6.4 Three-Talker Speech Separation

In Fig. D.5 we present the uPIT training progress as measured by MSE on the three-talker mixed speech training and validation sets WSJ0-3mix. We observe that similar to the two-talker scenario in Fig. D.4, a low training MSE is achieved, although the validation MSE is slightly higher. A better balance between the training and validation MSEs may be achieved by hyperparameter tuning. We also observe that increasing the model size decreases both training and validation MSE, which is expected due to the more variability in the dataset.

In Table D.7 we summarize the SDR improvement in dB from different uPIT separation configurations for three-talker mixed speech, in closed condition (CC) and open condition (OC). We observe that the basic uPIT-BLSTM model (896 units) compares favorably with DPCL++. Furthermore, with ad-

[5] [37, 39] did not use the SDR measure from [44]. Instead a related variant called scale-invariant SNR was used.

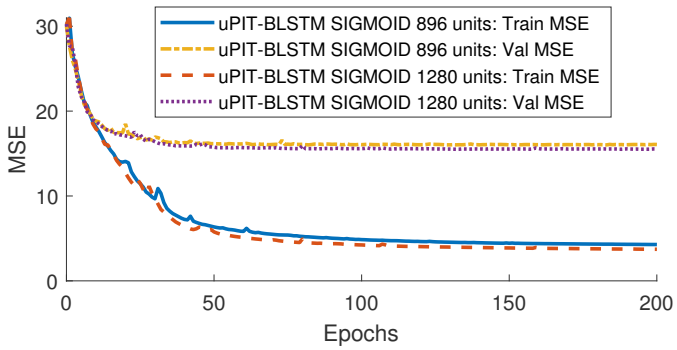


Fig. D.5: MSE over epochs on the WSJ0-3mix NPSM training and validation sets wit uPIT.

Table D.7: SDR improvements (dB) for different separation methods on the WSJ0-3mix dataset. ‡ indicates curriculum training.

Method	Units/ layer	Activation function	Opt. Assign.		Def. Assign.	
			CC	OC	CC	OC
Oracle NMF [36]	-	-	4.5	-	-	-
DPCL++‡ [39]	-	-	-	-	-	7.1
DANet [40]	-	-	-	-	-	7.7
DANet‡ [37]	-	-	-	-	-	**8.8**
uPIT-BLSTM	896	Sigmoid	10.0	9.9	7.4	7.2
uPIT-BLSTM	1280	Sigmoid	10.1	10.0	7.5	7.4
uPIT-BLSTM-RD	1280	Sigmoid	10.2	10.1	7.6	7.4
uPIT-BLSTM-ST	1280	Sigmoid	**10.7**	**10.6**	**7.9**	7.7
IRM	-	-	12.6	12.8	12.6	12.8
IPSM	-	-	15.1	15.3	15.1	15.3

‡ indicates curriculum training.

ditional units, further training and two-stage models (based on uPIT-BLSTM), uPIT achieves higher SDR than DPCL++ and similar SDR as DANet, without curriculum training, on this three-talker separation task.

6.5 Combined Two- and Three-Talker Speech Separation

To illustrate the flexibility of uPIT, we summarize in Table D.8 the performance of the three-speaker uPIT-BLSTM, and uPIT-BLSTM-ST models (from Table D.7), when they are trained and tested on both the WSJ0-2mix and WSJ0-3mix datasets, i.e. on both two- and three-speaker mixtures.

To be able to train the three-speaker models with the two-speaker WSJ0-2mix dataset, we extended WSJ0-2mix with a third "silent" channel. The silent channel consists of white Gaussian noise with an energy level 70 dB below the average energy level of the remaining two speakers in the mixture. When

Table D.8: SDR improvements (dB) for three-speaker models trained on both the WSJ0-2mix and WSJ0-3mix PSM datasets.

Method	2 Spkr.		3 Spkr.	
	Def. Assign.		Def. Assign.	
	CC	OC	CC	OC
uPIT-BLSTM	9.4	9.3	7.2	7.1
uPIT-BLSTM-ST	**10.2**	**10.1**	**8.0**	**7.8**
IRM	12.4	12.7	12.6	12.8
IPSM	14.9	15.1	15.1	15.3

Both models have 1280 units per layer and ReLU outputs.

we evaluated the model, we identified the two speaker-active output streams as the ones corresponding to the signals with the most energy.

We see from Table D.8 that uPIT-BLSTM achieves good, but slightly worse, performance compared to the corresponding two-speaker (Table D.6) and three-speaker (Table D.7) models. Surprisingly, the uPIT-BLSTM-ST model outperforms both the two-speaker (Table D.3) and three-speaker uPIT-BLSTM-ST (Table D.7) models. These results indicate that a single model can handle a varying, and more importantly, unknown number of speakers, without compromising performance. This is of great practical importance, since *a priori* knowledge about the number of speakers is not needed at test time, as required by competing methods such as DPCL++ [39] and DANet [37, 40].

During evaluation of the 3000 mixtures in the WSJ0-2mix test set, output stream one and two were the output streams with the most energy, i.e. the speaker-active output streams, in 2999 cases. Furthermore, output stream one and two had, on average, an energy level approximately 33 dB higher than the silent channel, indicating that the models successfully keep a constant permutation of the output masks throughout the test utterance. As an example, Fig. D.6 shows the spectrogram for a single two-speaker (male-vs-female) test case along with the spectrograms of the three output streams of the uPIT-BLSTM model, as well as the clean speech signals from each of the two speakers. Clearly, output streams one and two contain the most energy and output stream three consists primarily of a low energy signal without any clear structure. Furthermore, by comparing the spectrograms of the clean speech signals ("Speaker 1" and "Speaker 2" in Fig. D.6) to the spectrogram of the corresponding output streams, it is observed that they share many similarities, which indicate that the model kept a constant output-mask permutation for the entire mixture and successfully separated the two speakers into two separate output streams. This is also supported by the SDR improvements, which for output stream one ("Speaker 1") is 13.7 dB, and for output stream two ("Speaker 2") is 12.1 dB.

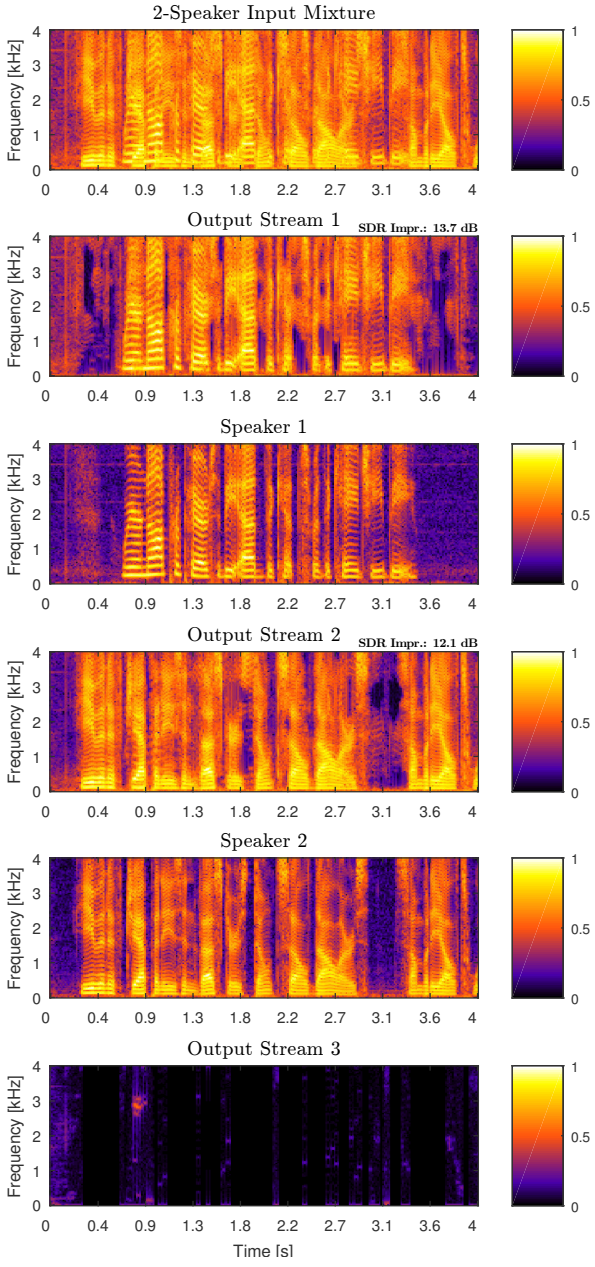


Fig. D.6: Spectrograms showing how a three-speaker BLSTM model trained with uPIT can separate a two-speaker mixture while keeping a constant output-mask permutation. The energy in output stream three is 63 dB lower than the energy in output stream one and two.

7 Conclusion and Discussion

In this paper, we have introduced the utterance-level Permutation Invariant Training (uPIT) technique for speaker independent multi-talker speech separation. We consider uPIT an interesting step towards solving the important cocktail party problem in a real-world setup, where the set of speakers is unknown during the training time.

Our experiments on two- and three-talker mixed speech separation tasks indicate that uPIT can indeed effectively deal with the label permutation problem. These experiments show that bi-directional Long Short-Term Memory (LSTM) Recurrent Neural Networks (RNNs) perform better than uni-directional LSTMs and Phase Sensitive Masks (PSMs) are better training criteria than Amplitude Masks (AM). Our results also suggest that the acoustic cues learned by the model are largely speaker and language independent since the models generalize well to unseen speakers and languages. More importantly, our results indicate that uPIT trained models do not require *a priori* knowledge about the number of speakers in the mixture. Specifically, we show that a single model can handle both two-speaker and three-speaker mixtures. This indicates that it might be possible to train a universal speech separation model using speech in various speaker, language and noise conditions.

The proposed uPIT technique is algorithmically simpler yet performs on par with DPCL [36, 39] and comparable to DANets [37, 40], both of which involve separate embedding and clustering stages during inference. Since uPIT, as a training technique, can be easily integrated and combined with other advanced techniques such as complex-domain separation and multi-channel techniques, such as beamforming, uPIT has great potential for further improvement.

Acknowledgment

We would like to thank Dr. John Hershey at MERL and Zhuo Chen at Columbia University for sharing the WSJ0-2mix and WSJ0-3mix datasets and for valuable discussions. We also thank Dr. Hakan Erdogan at Microsoft Research for discussions on PSM.

References

[1] S. Haykin and Z. Chen, "The Cocktail Party Problem," *Neural Comput.*, vol. 17, no. 9, pp. 1875–1902, 2005.

[2] A. W. Bronkhorst, "The Cocktail Party Phenomenon: A Review of Research on Speech Intelligibility in Multiple-Talker Conditions," *Acta Acust united Ac*, vol. 86, no. 1, pp. 117–128, 2000.

[3] E. C. Cherry, "Some Experiments on the Recognition of Speech, with One and with Two Ears," *J. Acoust. Soc. Am.*, vol. 25, no. 5, pp. 975–979, Sep. 1953.

[4] M. Cooke, J. R. Hershey, and S. J. Rennie, "Monaural Speech Separation and Recognition Challenge," *Comput. Speech Lang.*, vol. 24, no. 1, pp. 1–15, Jan. 2010.

[5] P. Divenyi, *Speech Separation by Humans and Machines*. Springer, 2005.

[6] D. P. W. Ellis, "Prediction-driven computational auditory scene analysis," Ph.D. dissertation, Massachusetts Institute of Technology, 1996.

[7] M. Cooke, *Modelling Auditory Processing and Organisation*. Cambridge University Press, 2005.

[8] D. Wang and G. J. Brown, *Computational Auditory Scene Analysis: Principles, Algorithms, and Applications*. Wiley-IEEE Press, 2006.

[9] Y. Shao and D. Wang, "Model-based sequential organization in cochannel speech," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 14, no. 1, pp. 289–298, Jan. 2006.

[10] K. Hu and D. Wang, "An Unsupervised Approach to Cochannel Speech Separation," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 21, no. 1, pp. 122–131, Jan. 2013.

[11] M. N. Schmidt and R. K. Olsson, "Single-Channel Speech Separation using Sparse Non-Negative Matrix Factorization," in *Proc. INTERSPEECH*, 2006, pp. 2614–2617.

[12] P. Smaragdis, "Convolutive Speech Bases and Their Application to Supervised Speech Separation," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 15, no. 1, pp. 1–12, Jan. 2007.

[13] J. L. Roux, F. Weninger, and J. R. Hershey, "Sparse NMF – half-baked or well done?" Mitsubishi Electric Research Labs (MERL), Tech. Rep. TR2015-023, 2015.

[14] D. D. Lee and H. S. Seung, "Algorithms for Non-negative Matrix Factorization," in *NIPS*, 2000, pp. 556–562.

[15] T. T. Kristjansson, J. R. Hershey, P. A. Olsen, S. J. Rennie, and R. A. Gopinath, "Super-human multi-talker speech recognition: the IBM 2006 speech separation challenge system," in *Proc. INTERSPEECH*, 2006, pp. 97–100.

[16] T. Virtanen, "Speech Recognition Using Factorial Hidden Markov Models for Separation in the Feature Space," in *Proc. INTERSPEECH*, 2006.

[17] M. Stark, M. Wohlmayr, and F. Pernkopf, "Source-Filter-Based Single-Channel Speech Separation Using Pitch Information," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 19, no. 2, pp. 242–255, Feb. 2011.

[18] Z. Ghahramani and M. I. Jordan, "Factorial Hidden Markov Models," *Machine Learning*, vol. 29, no. 2-3, pp. 245–273, 1997.

[19] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016. [Online]. Available: http://www.deeplearningbook.org

[20] D. Yu, L. Deng, and G. E. Dahl, "Roles of Pre-Training and Fine-Tuning in Context-Dependent DBN-HMMs for Real-World Speech Recognition," in *NIPS Workshop on Deep Learning and Unsupervised Feature Learning*, 2010.

[21] G. E. Dahl, D. Yu, L. Deng, and A. Acero, "Context-Dependent Pre-Trained Deep Neural Networks for Large-Vocabulary Speech Recognition," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 20, no. 1, pp. 30–42, Jan. 2012.

[22] F. Seide, G. Li, and D. Yu, "Conversational speech transcription using context-dependent deep neural networks," in *Proc. INTERSPEECH*, 2011, pp. 437–440.

[23] G. Hinton *et al.*, "Deep Neural Networks for Acoustic Modeling in Speech Recognition: The Shared Views of Four Research Groups," *IEEE Sig. Process. Mag.*, vol. 29, no. 6, pp. 82–97, Nov. 2012.

[24] W. Xiong *et al.*, "Achieving Human Parity in Conversational Speech Recognition," *arXiv:1610.05256 [cs]*, 2016.

[25] G. Saon *et al.*, "English Conversational Telephone Speech Recognition by Humans and Machines," *arXiv:1703.02136 [cs]*, 2017.

[26] Y. Wang and D. Wang, "Towards Scaling Up Classification-Based Speech Separation," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 21, no. 7, pp. 1381–1390, Jul. 2013.

[27] Y. Wang, A. Narayanan, and D. Wang, "On Training Targets for Supervised Speech Separation," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 22, no. 12, pp. 1849–1858, 2014.

[28] Y. Xu, J. Du, L.-R. Dai, and C.-H. Lee, "An Experimental Study on Speech Enhancement Based on Deep Neural Networks," *IEEE Sig. Process. Let.*, vol. 21, no. 1, pp. 65–68, Jan. 2014.

[29] F. Weninger *et al.*, "Speech Enhancement with LSTM Recurrent Neural Networks and Its Application to Noise-Robust ASR," in *LVA/ICA*. Springer, 2015, pp. 91–99.

[30] P.-S. Huang, M. Kim, M. Hasegawa-Johnson, and P. Smaragdis, "Joint Optimization of Masks and Deep Recurrent Neural Networks for Monaural Source Separation," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 23, no. 12, pp. 2136–2147, 2015.

[31] J. Chen, Y. Wang, S. E. Yoho, D. Wang, and E. W. Healy, "Large-scale training to increase speech intelligibility for hearing-impaired listeners in novel noises," *J. Acoust. Soc. Am.*, vol. 139, no. 5, pp. 2604–2612, 2016.

[32] M. Kolbæk, Z. H. Tan, and J. Jensen, "Speech Intelligibility Potential of General and Specialized Deep Neural Network Based Speech Enhancement Systems," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 25, no. 1, pp. 153–167, 2017.

[33] J. Du, Y. Tu, Y. Xu, L. Dai, and C. H. Lee, "Speech separation of a target speaker based on deep neural networks," in *ICSP*, 2014, pp. 473–477.

[34] T. Goehring, F. Bolner, J. J. M. Monaghan, B. van Dijk, A. Zarowski, and S. Bleeck, "Speech enhancement based on neural networks improves speech intelligibility in noise for cochlear implant users," *Hearing Research*, vol. 344, pp. 183–194, 2017.

[35] C. Weng, D. Yu, M. L. Seltzer, and J. Droppo, "Deep Neural Networks for Single-Channel Multi-Talker Speech Recognition," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 23, no. 10, pp. 1670–1679, 2015.

[36] J. R. Hershey, Z. Chen, J. L. Roux, and S. Watanabe, "Deep clustering: Discriminative embeddings for segmentation and separation," in *Proc. ICASSP*, 2016, pp. 31–35.

[37] Z. Chen, Y. Luo, and N. Mesgarani, "Deep attractor network for single-microphone speaker separation," in *Proc. ICASSP*, 2017, pp. 246–250.

[38] D. Yu, M. Kolbæk, Z.-H. Tan, and J. Jensen, "Permutation Invariant Training of Deep Models for Speaker-Independent Multi-talker Speech Separation," in *Proc. ICASSP*, 2017, pp. 241–245.

[39] Y. Isik, J. L. Roux, Z. Chen, S. Watanabe, and J. R. Hershey, "Single-Channel Multi-Speaker Separation Using Deep Clustering," in *Proc. INTERSPEECH*, 2016, pp. 545–549.

[40] Z. Chen, "Single Channel Auditory Source Separation with Neural Network," Ph.D., Columbia University, United States – New York, 2017.

[41] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput*, vol. 9, no. 8, pp. 1735–1780, 1997.

[42] D. S. Williamson, Y. Wang, and D. Wang, "Complex Ratio Masking for Monaural Speech Separation," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 24, no. 3, pp. 483–492, Mar. 2016.

[43] H. Erdogan, J. R. Hershey, S. Watanabe, and J. L. Roux, "Deep Recurrent Networks for Separation and Recognition of Single Channel Speech in Non-stationary Background Audio," in *New Era for Robust Speech Recognition: Exploiting Deep Learning*. Springer, 2017.

[44] E. Vincent, R. Gribonval, and C. Fevotte, "Performance measurement in blind audio source separation," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 14, no. 4, pp. 1462–1469, 2006.

[45] H. Erdogan, J. R. Hershey, S. Watanabe, and J. L. Roux, "Phase-sensitive and recognition-boosted speech separation using deep recurrent neural networks," in *Proc. ICASSP*, 2015, pp. 708–712.

[46] Y. Tu, J. Du, Y. Xu, L. Dai, and C. H. Lee, "Deep neural network based speech separation for robust speech recognition," in *ICSP*, 2014, pp. 532–536.

[47] D. Yu, K. Yao, and Y. Zhang, "The Computational Network Toolkit," *IEEE Sig. Process. Mag.*, vol. 32, no. 6, pp. 123–126, Nov. 2015.

[48] A. Agarwal *et al.*, "An introduction to computational networks and the computational network toolkit," Microsoft Technical Report {MSR-TR}-2014-112, Tech. Rep., 2014.

[49] A. Rix, J. Beerends, M. Hollier, and A. Hekstra, "Perceptual evaluation of speech quality (PESQ)-a new method for speech quality assessment of telephone networks and codecs," in *Proc. ICASSP*, vol. 2, 2001, pp. 749–752.

[50] J. Garofolo, D. Graff, P. Doug, and D. Pallett, "CSR-I (WSJ0) Complete LDC93s6a," 1993, philadelphia: Linguistic Data Consortium.

[51] Y. Gal and Z. Ghahramani, "A Theoretically Grounded Application of Dropout in Recurrent Neural Networks," *arXiv:1512.05287*, Dec. 2015.

[52] X. L. Zhang and D. Wang, "A Deep Ensemble Learning Method for Monaural Speech Separation," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 24, no. 5, pp. 967–977, May 2016.

[53] S. Nie, H. Zhang, X. Zhang, and W. Liu, "Deep stacking networks with time series for speech separation," in *Proc. ICASSP*, 2014, pp. 6667–6671.

[54] Z.-Q. Wang and D. Wang, "Recurrent Deep Stacking Networks for Supervised Speech Separation," in *Proc. ICASSP*, 2017, pp. 71–75.

[55] Y. Bengio, J. Louradour, R. Collobert, and J. Weston, "Curriculum Learning," in *ICML*, 2009, pp. 41–48.

This page intentionally left blank.

Paper E

Joint Separation and Denoising of Noisy Multi-Talker Speech Using Recurrent Neural Networks and Permutation Invariant Training

Morten Kolbæk, Dong Yu, Zheng-Hua Tan, and Jesper Jensen

The paper has been published in
Proceedings IEEE International Workshop on Machine Learning for Signal Processing, pp. 1-6, September 2017.

© 2017 IEEE
The layout has been revised.

Abstract

In this paper we propose to use utterance-level Permutation Invariant Training (uPIT) for speaker independent multi-talker speech separation and denoising, simultaneously. Specifically, we train deep bi-directional Long Short-Term Memory (LSTM) Recurrent Neural Networks (RNNs) using uPIT, for single-channel speaker independent multi-talker speech separation in multiple noisy conditions, including both synthetic and real-life noise signals. We focus our experiments on generalizability and noise robustness of models that rely on various types of a priori *knowledge e.g. in terms of noise type and number of simultaneous speakers.*

We show that deep bi-directional LSTM RNNs trained using uPIT in noisy environments can improve the Signal-to-Distortion Ratio (SDR) as well as the Extended Short-Time Objective Intelligibility (ESTOI) measure, on the speaker independent multi-talker speech separation and denoising task, for various noise types and Signal-to-Noise Ratios (SNRs). Specifically, we first show that LSTM RNNs can achieve large SDR and ESTOI improvements, when evaluated using known noise types, and that a single model is capable of handling multiple noise types with only a slight decrease in performance. Furthermore, we show that a single LSTM RNN can handle both two-speaker and three-speaker noisy mixtures, without a priori *knowledge about the exact number of speakers. Finally, we show that LSTM RNNs trained using uPIT generalize well to noise types not seen during training.*

1 Introduction

Focusing ones auditory attention towards a single speaker in a complex acoustic environment with multiple speakers and noise sources, is a task that humans are extremely good at [1]. However, achieving similar performance with machines has so far not been possible [2], although it would be highly desirable for a vast range of applications, such as mobile communications, robotics, hearing aids, speaker verification systems, etc.

Traditionally, speech denoising [3–8] and multi-talker speech separation [9–15] have been considered as two separate tasks in the literature, although, for many applications both speech separation and denoising are desired. For example, in a human-machine interface the machine must be able to identify what is being said, and by who, before it can decide which signal to focus on, and consequently respond and act upon.

The recent success of Deep Learning [16] has revolutionized a large number of scientific fields, and is currently achieving state-of-the-art results on topics ranging from medical diagnosis [17, 18] to Automatic Speech Recognition (ASR) [19, 20]. Also the area of single-channel speech enhancement has seen improvement, with deep learning algorithms that have been reported to improve speech intelligibility for normal hearing, hearing impaired and

cochlear implant users [8, 21–23]. Speaker independent multi-talker speech separation, on the other hand, has so far not taken a similar leap forward, partly due to the long-lasting label permutation problem (further described in Section 3), which has prevented progress on deep learning based techniques for this task.

Recently, two technical directions have been proposed for speaker independent multi-talker speech separation; a clustering based approach [11–13], and a regression based approach [10, 24]. The clustering based approaches include the Deep Clustering (DPCL) techniques [11, 12] and the DANet technique [13]. The regression based approaches include the Permutation Invariant Training (PIT) technique [10] and the utterance-level PIT (uPIT) technique [24]. The general idea behind the DPCL and DANet techniques is that the mixture signal can be represented in an embedding space, e.g. using Recurrent Neural Networks (RNNs), where the different source signals in the mixture form clusters. These clusters are then identified using a clustering technique, such as K-means. The clustering based techniques have shown impressive performance on two-speaker and three-speaker mixtures. The regression based PIT and uPIT techniques, which are described in detail in Section 3 utilize a cost function that jointly optimizes the label assignment and regression error end-to-end, hence effectively solving the label permutation problem.

Both clustering based and regression based methods [10–13, 23, 24] focus on ideal, noise-free training/testing conditions; i.e. situations where the mixtures contain clean speech only. For any practical application, background noise, e.g. due to interfering sound sources or non-ideal microphones must be expected. However, it is yet to be known how these techniques perform, when tested in noisy conditions that reflect a realistic usage scenario.

In this paper we apply the recently proposed uPIT technique [24] for speaker independent multi-talker speech separation and denoising, simultaneously. Specifically, we train deep bi-directional Long Short-Term Memory (LSTM) RNNs using uPIT for speaker independent multi-talker speech separation in multiple noisy conditions, including both synthetic and real-life, known and unknown, noise signals at various Signal-to-Noise Ratios (SNRs).

To the authors knowledge, this is the first attempt to perform speech separation and denoising simultaneously in a deep learning framework; hence, no competing baseline has been identified for this particular task.

2 Source Separation Using Deep Learning

The goal of single-channel speech separation is to separate a mixture of multiple speakers into the individual speakers using a single microphone record-

ing. Similarly, single-channel speech denoising aims to extract a single target speech signal from a noisy single channel recording.

Let $x_s[n]$, $n = 1, 2, \ldots, N$, $s = 1, 2, \ldots, S$ be the time domain source signal of length N from source s and let the observed mixture signal be defined as

$$y[n] = \sum_{s=1}^{S} x_s[n], \tag{E.1}$$

where $x_1[n]$ is a speech signal and $x_s[n]$, $s = 2, \ldots, S$ can be either speech or additive noise signals. Furthermore, let $X_s(i, f)$ and $Y(i, f)$, $i = 1, \ldots, K$, $f = 1, 2, \ldots, L$ be the L-point Short-Time discrete Fourier Transforms (STFT) of $x_s[n]$ and $y[n]$, respectively. Also, let $\mathbf{x}_{s,i} = \Big[X_s(i,1), X_s(i,2), \ldots, X_s(i, \frac{L}{2}+1)\Big]^T \in \mathbb{C}^{\frac{L}{2}+1}$ and $\mathbf{y}_i = \Big[Y(i,1), Y(i,2), \ldots, Y(i, \frac{L}{2}+1)\Big]^T \in \mathbb{C}^{\frac{L}{2}+1}$ denote the single-sided STFT spectrum, at frame i, for sources $s = 1, 2, \ldots, S$ and the mixture signal, respectively.

We define the magnitudes of the source signals and mixture signal as $A_s(i, f) \triangleq |X_s(i, f)|$ and $R(i, f) \triangleq |Y(i, f)|$, respectively, and their corresponding single-sided magnitude spectra as $\mathbf{a}_{s,i} = \Big[A_s(i,1), \ldots, A_s(i, \frac{L}{2}+1)\Big]^T \in \mathbb{R}^{\frac{L}{2}+1}$ and $\mathbf{r}_i = \Big[R(i,1),\ R(i,2),\ \ldots,\ R(i, \frac{L}{2}+1)\Big]^T \in \mathbb{R}^{\frac{L}{2}+1}$. For separating the mixture signal $\mathbf{y}_i$ into estimated target signal magnitudes $\mathbf{a}_{s,i}$, $s = 1, 2, \ldots, S$, we adopt the approach from [24] and estimate a set of masks $M_s(t, f)$, $s = 1, 2, \ldots, S$ using bi-directional LSTM RNNs.

Let $\mathbf{m}_{s,i} = \Big[M_s(i,1)\ ,\ M_s(i,2)\ ,\ \ldots,\ M_s(i, \frac{L}{2}+1)\Big]^T \in \mathbb{R}^{\frac{L}{2}+1}$ be the ideal mask (to be defined in Sec. 2.1) for speaker s at frame i. The masks $\mathbf{m}_{s,i}$, $s = 1, 2, \ldots, S$ are then used to extract the target signal magnitudes as $\mathbf{a}_{s,i} = \mathbf{m}_{s,i} \circ \mathbf{r}_i$, $s = 1, 2, \ldots, S$, $i = 1, 2, \ldots, K$ where $\circ$ is the element-wise product, i.e. the Hadamard product. Similarly, when the masks are estimated by a deep learning model we arrive at the estimated signal magnitudes as $\hat{\mathbf{a}}_{s,i} = \hat{\mathbf{m}}_{s,i} \circ \mathbf{r}_i$, $s = 1, 2, \ldots, S$, $i = 1, 2, \ldots, K$. The overlap-and-add technique and the inverse discrete Fourier transform, using the phase of the mixture signal, is used for reconstructing $\hat{\mathbf{a}}_{s,i}$, $i = 1, 2, \ldots, K$ in the time domain.

2.1 Mask Estimation and Loss functions

A large number of training targets and loss functions have been proposed for masking based source separation [7, 23, 25]. Since the one reasonable goal is to have an accurate reconstruction, a loss function based on the reconstruction error instead of the mask estimation error is preferable [23].

In [24], different such loss functions were investigated for speaker independent multi-talker speech separation and the best performing one was

found to be the Phase Sensitive Approximation (PSA) loss function [7], which for frame i is given as

$$\begin{aligned} J_i^{PSA} &= \sum_{s=1}^{S} \|\hat{\mathbf{m}}_{s,i} \circ \mathbf{r}_i - \mathbf{a}_{s,i} \cos(\phi_{s,i})\|_2^2 \\ &= \sum_{s=1}^{S} \|\hat{\mathbf{a}}_{s,i} - \mathbf{a}_{s,i} \cos(\phi_{s,i})\|_2^2, \end{aligned} \tag{E.2}$$

where $\phi_{s,i} = \phi_{y,i} - \phi_{s,i}$ is the element-wise phase difference between the mixture $\mathbf{y}_i$ and the source $\mathbf{x}_{s,i}$ and $||\cdot||_2$ is the ℓ^2-norm.

In contrast to the classical squared error loss function, i.e. Eq. (E.2) without the cosine term, the PSA loss function accounts for some of the errors introduced by the noisy phase used in the reconstruction. When the PSA loss function is used for mask estimation, the actual mask estimated is the Ideal Phase Sensitive Filter (IPSF) [7], which due to the phase correction property is preferable over other commonly used masks such as the ideal ratio mask, or the ideal amplitude mask [23].

3 Permutation Invariant Training

Permutation Invariant Training (PIT) is a generalization of the traditional approach for training Deep Neural Networks (DNNs) for regression based source separation problems, such as speaker separation or denoising.

For training a DNN based source separation model with S output masks, $\hat{\mathbf{m}}_{s,i}$, $s = 1, \ldots, S$, an MSE criterion is typically used and is computed between the true sources $\mathbf{a}_{s,i}$ and the estimated sources $\hat{\mathbf{a}}_{s,i} = \hat{\mathbf{m}}_{s,i} \circ \mathbf{r}_i$, $s = 1, \ldots, S$, $i = 1, \ldots, K$. However, with multiple outputs, it is not trivial to pair the outputs with the correct targets. The commonly used approach for pairing a given output $\hat{\mathbf{a}}_{s,i}$ to a certain target $\mathbf{a}_{s,i}$ is to predefine the targets into an ordered list, such that output one is always paired with e.g. target one, i.e. $(\mathbf{a}_{1,i}, \hat{\mathbf{a}}_{1,i})$, output two with target two $(\mathbf{a}_{2,i}, \hat{\mathbf{a}}_{2,i})$, etc.

For tasks such as speech denoising with a single speaker in noise, or speech separation of known speakers[15], simply predefining the ordering of the targets works well and the DNN can learn to correctly separate the sources and will provide the correct source at the output corresponding to the correct target. However, for mixtures containing similar signals, such as unknown equal energy male speakers, this standard training approach fails to converge [10, 11, 14]. Empirically, it is found that DNNs are likely to change permutation from one frame to another for highly similar sources. Hence, predefining the ordering of the targets, might not be the optimal solution, and clearly a bad solution for certain types of signals. This phenomenon, and the

challenge of choosing the output-target permutation during training, is commonly known as the label permutation or ambiguity problem [11, 13, 14, 24].

In [10] a solution to the label permutation problem was proposed, where targets are provided as a set instead of an ordered list and the output-target permutation θ, for a given frame, is defined as the permutation that minimizes the cost function in question (e.g. squared error) over all possible permutations $\mathcal{P}$. Following this approach combined with the PSA loss function, a permutation invariant training criterion and corresponding error J_i^{PIT}, for the i^{th} frame, can be formulated as

$$J_i^{PIT} = \min_{\theta \in \mathcal{P}} \sum_{s=1}^{S} \|\hat{\mathbf{a}}_{s,i} - \mathbf{a}_{\theta(s),i} \cos(\phi_{\theta(s),i})\|_2^2. \tag{E.3}$$

As shown in [10], Eq. (E.3) effectively solves the label permutation problem. However, since PIT as defined in Eq. (E.3) operates on frames, the DNN only learns to separate the input mixtures into sources at the frame level, and not the utterance level. In practice, this means that the mixture might be correctly separated, but the frames belonging to a particular speaker are not assigned the same output index throughout the utterance and without exact knowledge about the speaker-output permutation, it is very difficult to correctly reconstruct the separated sources. In order to have the sources separated at the utterance-level, so that all frames from a particular output belong to the same source, additional speaker tracing or very large input-output contexts are needed [10].

3.1 Utterance-Level Permutation Invariant Training

In [24] an extension to PIT, known as utterance-level PIT (uPIT) was proposed for solving the speaker-output permutation problem. In uPIT, the output-target permutation θ is given as the permutation that gives the minimum squared error over all possible permutations for the entire utterance, instead of only a single frame. Formally, the utterance-level permutation used for training is found as

$$\theta^* = \operatorname*{argmin}_{\theta \in \mathcal{P}} \sum_{s=1}^{S} \sum_{i=1}^{K} \|\hat{\mathbf{a}}_{s,i} - \mathbf{a}_{\theta(s),i} \cos(\phi_{\theta(s),i})\|_2^2, \tag{E.4}$$

and the permutation θ^* is then used *for all* frames within the current utterance, hence an utterance-level loss $J_{\theta^*,i}^{uPIT}$ for the i^{th} frame in a given utterance is defined as

$$J_{\theta^*,i}^{uPIT} = \sum_{s=1}^{S} \|\hat{\mathbf{a}}_{s,i} - \mathbf{a}_{\theta^*(s),i} \cos(\phi_{\theta^*(s),i})\|_2^2. \tag{E.5}$$

Using the same permutation *for all* frames in the entire utterance has the consequence that the smallest per-frame error will not always be used for

training as with original PIT. Instead the smallest *per-utterance* error will be used, which enforces the estimated sources to stay at the same DNN outputs for the entire utterance. Ideally, this means that each DNN output contains a single source. Finally, since the whole utterance is needed for computing the utterance-level permutation in Eq. (E.4), RNNs are a natural choice of DNN model for this loss function.

4 Experimental Design

To study the noise robustness of the uPIT technique, we have conducted several experiments with noise corrupted mixtures of multiple speakers. Since uPIT uses the noise-free source signals as training targets, a denoising capability is already present in the uPIT framework. By simply adding noise to the multi-speaker input mixture, a model trained with uPIT will not only learn to separate the sources but also to remove the noise.

4.1 Noise-Free Multi-Talker Speech Mixtures

We have used the noise-free two-speaker mixture (WSJ0-2mix) and three-speaker mixture (WSJ0-3mix)[1] datasets for all experiments conducted in this paper. These datasets have been used in [10–12, 24], which allows us to relate the performance of uPIT in noisy conditions with the performance in noise-free conditions. The feature representation is based on 129-dimensional STFT magnitude spectra, extracted from a 256 point STFT using a sampling frequency of 8 kHz, a hanning window size of 32 ms and a 16 ms frame shift.

The WSJ0-2mix dataset was derived from the WSJ0 corpus [26]. The WSJ0-2mix training set and validation set contain two-speaker mixtures generated by randomly selecting pairs of utterances from 49 male and 51 female speakers from the WSJ0 training set entitled si_tr_s. The two utterances are then mixed with a difference in active speech level [27] uniformly chosen between 0 dB and 5 dB. The training and validation sets consist of 20000 and 5000 mixtures, respectively, which is equivalent to approximately 30 hours of training data and 5 hours of validation data. The test set was similarly generated using utterances from 16 speakers from the WSJ0 validation set si_dt_05 and evaluation set si_et_05, and consists of 5000 mixtures or approximately 5 hours of data. That is, the speakers in the test set are different from the speakers in the training and validation sets. The WSJ0-3mix dataset was generated using a similar approach but contains mixtures of speech from three speakers.

Since we want a single RNN architecture that can handle both two-speaker and three-speaker mixtures, we have chosen a model architecture with three

[1] Available at: http://www.merl.com/demos/deep-clustering

outputs. The specific architecture is described in detail in Sec. 4.3. To ensure that the model can handle both two-speaker and three-speaker mixtures, the model must be trained on both scenarios, so we have combined the WSJ0-2mix and WSJ0-3mix datasets into a larger WSJ0-2+3mix dataset. To allow this fusion, we have extended the WSJ0-2mix dataset with a third "silent" speaker, such that the combined WSJ0-2+3mix dataset consists of only three speaker mixtures, but half of the mixtures contain three speakers, and the remaining half contain two speaker mixtures (and a "silent speaker"). To minimize the risk of numerical issues, e.g. in computing ideal masks, the third "silent" speaker consists of white Gaussian noise with an average energy level 70 dB below the average energy of the other two speakers in the mixture.

4.2 Noisy Multi-Talker Speech Mixtures

To simulate noisy environments, we follow the common approach [3] for generating noisy mixtures with additive noise and simply add the noise-free WSJ0-2+3mix mixture signal with a noise signal. To achieve a certain SNR the noise signal is scaled based on the active speech level of the noise-free mixture signal as per ITU P.56 [27].

To evaluate the robustness of the uPIT model against a stationary noise type, we use a synthetic Speech Shaped Noise (SSN) signal. The SSN noise signal is constructed by filtering a Gaussian white noise sequence through a 12^{th}-order all-pole filter with coefficients found from linear predictive coding analysis of 100 randomly chosen TIMIT sentences [28].

To evaluate the robustness against a highly non-stationary noise type we use a synthetic 6-speaker Babble (BBL) noise. The BBL noise signal is also based on TIMIT. The corpus, which consists of a total of 6300 spoken sentences, is randomly divided into 6 groups of 1050 concatenated utterances. Each group is then normalized to unit energy and truncated to equal length followed by addition of the six groups. This results in a BBL noise sequence with a duration of over 50 min.

To evaluate the robustness against realistic noise types we use the street (STR), cafeteria (CAF), bus (BUS), and pedestrian (PED) noise signals from the CHiME3 dataset [29]. These noise signals are real-life recordings in their respective environments.

All six noise signals are divided into a 40 min. training sequence, a 5 min. validation sequence and a 5 min. test sequence. That is, the noise signals used for training and validation are different from the sequence used for testing.

Table E.1: Training conditions for different models.

Model ID	Dataset + Noise type (SNR: -5 dB – 10 dB)
LSTM1	WSJ0-2+3mix + SSN
LSTM2	WSJ0-2+3mix + BBL
LSTM3	WSJ0-2+3mix + STR
LSTM4	WSJ0-2+3mix + CAF
LSTM5	WSJ0-2+3mix + SSN + BBL + STR + CAF
LSTM6	WSJ0-2mix + BBL
LSTM7	WSJ0-3mix + BBL

4.3 Model Architectures and Training

For evaluating uPIT in noisy environments we have trained a total of seven bi-directional LSTM RNNs [30], using the training conditions, i.e. datasets and noise types, presented in Table. E.1. LSTM1-5 were trained on the WSJ0-2+3mix dataset, which contains a mix of both two-speaker and three-speaker mixtures. LSTM1-4 are noise type specific in the sense that they were trained using only a single noise type. LSTM5 was trained on all four noise types. LSTM6 and LSTM7 were trained using WSJ0-2mix and WSJ0-3mix datasets, respectively, and only a single noise type. LSTM5 will show the performance degradation, if any, when less *a priori* knowledge about the noise types is available. Similarly, LSTM6-7 will show the potential performance improvement if the number of speakers in the mixture is known *a priori*. Each mixture in the dataset was corrupted with noise at a specific SNR, uniformly chosen between -5 dB and 10 dB.

Each model has three bi-directional LSTM layers, and a fully-connected output layer with ReLU [16] activation functions. LSTM1-5 and LSTM7 have 1280 LSTM cells in each layer and LSTM6 has 896 cells, to be compliant with [24]. The input dimension is 129, i.e a single frame $\mathbf{r}_i$ and the output dimension is $3 \times 129 = 387$, i.e. $\hat{\mathbf{a}}_{s,i}$, $s = 1, 2, 3$. We apply 50% dropout [16] between the LSTM layers, and the outputs from the forward and backward LSTMs, from one layer, are concatenated before they are used as input to the subsequent layer. LSTM6 has approximately $46 \cdot 10^6$ trainable parameters, and LSTM1-5 and 7 have approximately $94 \cdot 10^6$ trainable parameters, which are found using stochastic gradient descent with gradients found by backpropagation. In all the experiments, the maximum number of epochs was set to 200 and the learning rates were set to $2 \cdot 10^{-5}$ per sample initially, and scaled down by 0.7 when the training cost increased on the training set. The training was terminated when the learning rate got below 10^{-10}. Each minibatch contains 8 randomly selected utterances. All models are implemented using the Microsoft Cognitive Toolkit (CNTK) [31][2].

[2] Available at: `https://www.cntk.ai/`

5 Experimental Results

We evaluated the noise robustness of LSTM1-7 using the Signal to Distortion Ratio (SDR) [32] and the Extended Short-Time Objective Intelligibility (ESTOI) measure [33]. The SDR is an often used performance metric for source separation and is defined in dB. The ESTOI measure estimates speech intelligibility and has been found to be highly correlated with human listening tests [33], especially for modulated maskers. The ESTOI measure is defined in the range $[-1, 1]$, and higher is better. When evaluating SDR and ESTOI, we choose the output-target permutation that maximizes the given performance metric. Furthermore, when evaluating two-speaker mixtures, we identify the silent speaker as the output with the least energy and then compute the performance metric based on the remaining two outputs.

Tables E.2 to E.5 summarize the SDR *improvements* achieved by LSTM1-5 on two and three-speaker mixtures corrupted by SSN, BBL, STR, and CAF noise, respectively. The improvements are relative to the SDR of the noisy mixture without processing ("No Proc." in Tables). Tables E.8 to E.11 summarize ESTOI *improvements* achieved by the same models in similar conditions. We evaluate the models at the challenging SNR of -5 dB, as well as at 0, 5 , and 20 dB. At an input SNR of -5 dB, speech intelligibility, as estimated by ESTOI, is severely degraded, primarily due to the noise component, whereas speech intelligibility degradation at 20 dB is primarily caused by the competing talkers in the mixture itself. As a reference, we also reported the IPSF performance, which uses oracle information and therefore serves as an upper performance bound on this particular task.

From Tables E.2 to E.5 and E.8 to E.11 we see that all noise-type specific models, i.e. LSTM1-4, in general achieve large SDR and ESTOI improvements with an average improvement of 9.1 dB and 0.18 for SDR and ESTOI, respectively, for two-speaker mixtures and 7.2 dB and 0.13, respectively, for three-speaker mixtures. Furthermore, we see that LSTM5 performs only slightly worse than the noise type specific models, which is interesting, since LSTM5 and LSMT1-4 have all been trained with 60 hours of speech, but LSTM5 have only seen 15 hours of each noise type, compared to 60 hours for LSTM1-4. We also observe that the highly non-stationary BBL noise seems to be considerably harder than the three other noise types, which corresponds well with existing literature [3, 34, 35].

Tables E.6 and E.12 summarize the performance of LSTM6 and LSTM7. We observe that both models perform approximately similar to the noise-type-general LSTM5. More surprisingly, we see that LSTM2 consistently outperforms both LSTM6 and LSTM7, which corresponds well with a similar observation in the noise-free case in [24]. These results are of great importance, since they show that training a model on noisy three-speaker mixtures

Table E.2: SDR improvements for LSTM1 and 5 tested on SSN.

	2-Speaker				3-Speaker			
SNR [dB]	No Proc.	IPSF	LSTM1	LSTM5	No Proc.	IPSF	LSTM1	LSTM5
-5	-8.8	15.9	9.6	9.4	-10.3	16.6	8.0	7.8
0	-5.1	14.5	9.1	9.0	-7.0	15.2	7.6	7.4
5	-2.4	13.9	8.6	8.4	-4.8	14.6	7.0	6.9
20	0.0	14.8	8.7	8.8	-3.0	15.1	6.6	6.7
Avg.	-4.1	14.8	9.0	8.9	-6.3	15.4	7.3	7.2

Table E.3: SDR improvements for LSTM2 and 5 tested on BBL.

	2-Speaker				3-Speaker			
SNR [dB]	No Proc.	IPSF	LSTM2	LSTM5	No Proc.	IPSF	LSTM2	LSTM5
-5	-8.9	17.2	6.0	5.4	-10.4	17.8	4.4	3.8
0	-5.1	15.4	8.1	7.6	-7.1	16.0	6.3	5.8
5	-2.4	14.5	8.5	8.1	-4.8	15.1	6.7	6.5
20	0.0	14.8	9.0	8.8	-3.0	15.2	6.8	6.7
Avg.	-4.1	15.5	7.9	7.5	-6.3	16.0	6.0	5.7

helps the model separating noisy two-speaker mixtures, and vice versa.

Tables E.7 and E.13 summarize the performance of LSTM5, when evaluated using speech mixtures corrupted with the two unknown noise types, PED and BUS, i.e. noise types not included in the training set. We see that LSTM5 achieves large SDR and ESTOI improvements for both noise types, at almost all SNRs. More importantly, we observe that the scores are comparable with, and in some cases even exceed, the performance of LSTM5, when it was evaluated using known noise types as reported in Tables E.2 to E.5 and E.8 to E.11. These results indicate that LSTM5 is relatively robust against variations in the noise distribution.

In general, we observe SDR improvements for all models that are comparable in magnitude with the noise-free case [10–12, 24]. However, the SDR measure, as well as ESTOI, do not differentiate between distortions from other speakers (such as source to inference ratio from [32]) and distortion from the noise source. This means that the trade-off between speech separation and noise-reduction is yet to be fully understood. We leave this topic for future research.

Table E.4: SDR improvements for LSTM3 and 5 tested on STR.

SNR [dB]	2-Speaker				3-Speaker			
	No Proc.	IPSF	LSTM3	LSTM5	No Proc.	IPSF	LSTM3	LSTM5
-5	-8.9	18.2	11.5	11.5	-10.4	18.6	9.7	9.6
0	-5.2	16.2	10.2	10.2	-7.1	16.7	8.4	8.3
5	-2.4	14.9	9.2	9.1	-4.8	15.5	7.3	7.2
20	0.0	14.9	8.9	8.8	-3.0	15.2	6.6	6.7
Avg.	-4.1	16.1	9.9	9.9	-6.3	16.5	8.0	7.9

Table E.5: SDR improvements for LSTM4 and 5 tested on CAF.

SNR [dB]	2-Speaker				3-Speaker			
	No Proc.	IPSF	LSTM4	LSTM5	No Proc.	IPSF	LSTM4	LSTM5
-5	-8.9	18.2	10.0	9.9	-10.4	18.6	8.4	8.2
0	-5.1	16.3	9.7	9.5	-7.1	16.8	7.9	7.7
5	-2.4	15.1	9.0	8.9	-4.8	15.6	7.1	6.9
20	0.0	14.8	8.8	8.8	-3.0	15.2	6.7	6.6
Avg.	-4.1	16.1	9.4	9.3	-6.3	16.6	7.5	7.3

Table E.6: SDR improvements for LSTM6, 7 and 5 tested on BBL.

SNR [dB]	2-Speaker				3-Speaker			
	No Proc.	IPSF	LSTM6	LSTM5	No Proc.	IPSF	LSTM7	LSTM5
-5	-8.9	17.2	5.6	5.4	-10.4	17.8	4.0	3.8
0	-5.1	15.4	7.7	7.6	-7.1	16.0	5.7	5.8
5	-2.4	14.5	8.0	8.1	-4.8	15.1	6.3	6.5
20	0.0	14.9	8.4	8.8	-3.0	15.2	6.4	6.7
Avg.	-4.1	15.5	7.4	7.5	-6.3	16.0	5.6	5.7

Table E.7: SDR improvements for LSTM5 tested on BUS and PED.

SNR [dB]	2-Speaker						3-Speaker					
	No Proc.		IPSF		LSTM5		No Proc.		IPSF		LSTM5	
	BUS	PED	BUS	PED	BUS	PED	BUS	PED	BUS	PED	BUS	PED
-5	-9.0	-8.9	19.6	16.7	11.7	7.3	-10.5	-10.4	19.9	17.4	9.7	5.7
0	-5.2	-5.2	17.3	14.9	10.7	7.8	-7.2	-7.1	17.6	15.7	8.5	6.3
5	-2.4	-2.4	15.7	14.1	9.5	7.9	-4.8	-4.8	16.1	14.8	7.4	6.3
20	0.0	0.0	14.9	14.8	8.8	8.7	-3.0	-3.0	15.2	15.2	6.7	6.7
Avg.	-4.1	-4.1	16.9	15.1	10.2	7.9	-6.4	-6.3	17.2	15.8	8.1	6.2

Table E.8: ESTOI improvements for LSTM1 and 5 tested on SSN.

	2-Speaker				3-Speaker			
SNR [dB]	No Proc.	IPSF	LSTM1	LSTM5	No Proc.	IPSF	LSTM1	LSTM5
-5	0.18	0.65	0.17	0.16	0.14	0.69	0.10	0.09
0	0.29	0.58	0.23	0.22	0.22	0.63	0.15	0.14
5	0.39	0.50	0.23	0.22	0.29	0.58	0.17	0.16
20	0.54	0.39	0.17	0.18	0.38	0.53	0.15	0.15
Avg.	0.35	0.53	0.20	0.20	0.26	0.61	0.14	0.14

Table E.9: ESTOI improvements for LSTM2 and 5 tested on BBL.

	2-Speaker				3-Speaker			
SNR [dB]	No Proc.	IPSF	LSTM2	LSTM5	No Proc.	IPSF	LSTM2	LSTM5
-5	0.19	0.66	0.09	0.06	0.14	0.70	0.04	0.02
0	0.29	0.59	0.18	0.15	0.22	0.65	0.11	0.09
5	0.39	0.51	0.21	0.20	0.29	0.60	0.15	0.14
20	0.53	0.40	0.19	0.18	0.37	0.53	0.15	0.15
Avg.	0.35	0.54	0.17	0.15	0.26	0.62	0.11	0.10

Table E.10: ESTOI improvements for LSTM3 and 5 tested on STR.

	2-Speaker				3-Speaker			
SNR [dB]	No Proc.	IPSF	LSTM3	LSTM5	No Proc.	IPSF	LSTM3	LSTM5
-5	0.24	0.60	0.16	0.15	0.18	0.65	0.10	0.09
0	0.32	0.54	0.21	0.19	0.24	0.61	0.14	0.13
5	0.40	0.49	0.21	0.20	0.30	0.57	0.15	0.15
20	0.54	0.39	0.18	0.18	0.37	0.53	0.15	0.15
Avg.	0.38	0.51	0.19	0.18	0.27	0.59	0.14	0.13

Table E.11: ESTOI improvements for LSTM4 and 5 tested on CAF.

	2-Speaker				3-Speaker			
SNR [dB]	No Proc.	IPSF	LSTM4	LSTM5	No Proc.	IPSF	LSTM4	LSTM5
-5	0.24	0.60	0.13	0.12	0.19	0.65	0.08	0.07
0	0.33	0.54	0.18	0.17	0.25	0.61	0.12	0.11
5	0.41	0.48	0.20	0.19	0.30	0.58	0.15	0.14
20	0.53	0.39	0.18	0.18	0.37	0.53	0.15	0.15
Avg.	0.38	0.50	0.17	0.17	0.28	0.59	0.12	0.12

Table E.12: ESTOI improvements for LSTM6, 7 and 5 tested on BBL.

	2-Speaker				3-Speaker			
SNR [dB]	No Proc.	IPSF	LSTM6	LSTM5	No Proc.	IPSF	LSTM7	LSTM5
-5	0.20	0.66	0.07	0.06	0.14	0.69	0.02	0.02
0	0.30	0.59	0.16	0.16	0.22	0.65	0.08	0.09
5	0.39	0.52	0.20	0.20	0.29	0.60	0.13	0.14
20	0.54	0.40	0.17	0.19	0.38	0.53	0.14	0.15
Avg.	0.36	0.54	0.15	0.15	0.26	0.62	0.09	0.10

Table E.13: ESTOI improvements for LSTM5 tested on BUS and PED.

	2-Speaker						3-Speaker					
SNR [dB]	No Proc.		IPSF		LSTM5		No Proc.		IPSF		LSTM5	
	BUS	PED	BUS	PED	BUS	PED	BUS	PED	BUS	PED	BUS	PED
-5	0.32	0.18	0.55	0.64	0.14	0.08	0.24	0.14	0.61	0.68	0.08	0.03
0	0.39	0.28	0.50	0.58	0.18	0.15	0.28	0.21	0.58	0.63	0.12	0.09
5	0.45	0.37	0.46	0.52	0.20	0.20	0.32	0.28	0.56	0.59	0.14	0.13
20	0.55	0.54	0.39	0.40	0.18	0.18	0.38	0.37	0.53	0.53	0.15	0.15
Avg.	0.43	0.34	0.47	0.54	0.17	0.15	0.31	0.25	0.57	0.61	0.12	0.10

6 Conclusion

In this paper we have proposed utterance-level Permutation Invariant Training (uPIT) for speaker independent multi-talker speech separation and denoising. Differently from prior works, that focus only on the ideal noise-free setting, we focus on the more realistic scenario of speech separation in noisy environments. Specifically, using the uPIT technique we have trained bi-directional Long Short-Term Memory (LSTM) Recurrent Neural Networks (RNNs), to separate two and three-speaker mixtures corrupted by multiple noise types at a wide range of Signal to Noise Ratios (SNRs).

We show that bi-directional LSTM RNNs trained with uPIT are capable of improving both Signal to Distortion Ratio (SDR), as well as the Extended Short-Time Objective Intelligibility (ESTOI) measure for challenging noise types and SNRs. Specifically, we show that LSTM RNNs achieve large SDR and ESTOI improvements, when evaluated using noise types seen during training, and that a single model is capable of handling multiple noise types with only a slight decrease in performance. Furthermore, we show that a single LSTM RNN can handle both two-speaker and three-speaker noisy mixtures, without *a priori* knowledge about the exact number of speakers. Finally, we show that LSTM RNNs trained using uPIT generalizes well to unknown noise types.

References

[1] A. W. Bronkhorst, "The Cocktail Party Phenomenon: A Review of Research on Speech Intelligibility in Multiple-Talker Conditions," *Acta Acust united Ac*, vol. 86, no. 1, pp. 117–128, 2000.

[2] J. H. McDermott, "The cocktail party problem," *Current Biology*, vol. 19, no. 22, pp. R1024–R1027, Dec. 2009.

[3] M. Kolbæk, Z. H. Tan, and J. Jensen, "Speech Intelligibility Potential of General and Specialized Deep Neural Network Based Speech Enhancement Systems," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 25, no. 1, pp. 153–167, 2017.

[4] J. Chen and D. Wang, "Long Short-Term Memory for Speaker Generalization in Supervised Speech Separation," in *Proc. INTERSPEECH*, 2016, pp. 3314 – 3318.

[5] F. Weninger, F. Eyben, and B. Schuller, "Single-channel speech separation with memory-enhanced recurrent neural networks," in *Proc. ICASSP*, 2014, pp. 3709–3713.

[6] F. Weninger, J. R. Hershey, J. L. Roux, and B. Schuller, "Discriminatively trained recurrent neural networks for single-channel speech separation," in *GlobalSIP*, 2014, pp. 577–581.

[7] H. Erdogan, J. R. Hershey, S. Watanabe, and J. L. Roux, "Phase-sensitive and recognition-boosted speech separation using deep recurrent neural networks," in *Proc. ICASSP*, 2015, pp. 708–712.

[8] J. Chen, Y. Wang, S. E. Yoho, D. Wang, and E. W. Healy, "Large-scale training to increase speech intelligibility for hearing-impaired listeners in novel noises," *J. Acoust. Soc. Am.*, vol. 139, no. 5, pp. 2604–2612, 2016.

[9] J. Du, Y. Tu, Y. Xu, L. Dai, and C. H. Lee, "Speech separation of a target speaker based on deep neural networks," in *ICSP*, 2014, pp. 473–477.

[10] D. Yu, M. Kolbæk, Z.-H. Tan, and J. Jensen, "Permutation Invariant Training of Deep Models for Speaker-Independent Multi-talker Speech Separation," in *Proc. ICASSP*, 2017, pp. 241–245.

[11] J. R. Hershey, Z. Chen, J. L. Roux, and S. Watanabe, "Deep clustering: Discriminative embeddings for segmentation and separation," in *Proc. ICASSP*, 2016, pp. 31–35.

[12] Y. Isik, J. L. Roux, Z. Chen, S. Watanabe, and J. R. Hershey, "Single-Channel Multi-Speaker Separation Using Deep Clustering," in *Proc. INTERSPEECH*, 2016, pp. 545–549.

[13] Z. Chen, Y. Luo, and N. Mesgarani, "Deep attractor network for single-microphone speaker separation," in *Proc. ICASSP*, 2017, pp. 246–250.

[14] C. Weng, D. Yu, M. L. Seltzer, and J. Droppo, "Deep Neural Networks for Single-Channel Multi-Talker Speech Recognition," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 23, no. 10, pp. 1670–1679, 2015.

[15] P.-S. Huang, M. Kim, M. Hasegawa-Johnson, and P. Smaragdis, "Joint Optimization of Masks and Deep Recurrent Neural Networks for Monaural Source Separation," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 23, no. 12, pp. 2136–2147, 2015.

[16] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016.

[17] V. Gulshan *et al.*, "Development and Validation of a Deep Learning Algorithm for Detection of Diabetic Retinopathy in Retinal Fundus Photographs," *JAMA*, vol. 316, no. 22, pp. 2402–2410, 2016.

[18] A. Esteva *et al.*, "Dermatologist-level classification of skin cancer with deep neural networks," *Nature*, vol. 542, no. 7639, pp. 115–118, 2017.

[19] W. Xiong *et al.*, "Achieving Human Parity in Conversational Speech Recognition," *arXiv:1610.05256 [cs]*, 2016.

[20] G. Saon *et al.*, "English Conversational Telephone Speech Recognition by Humans and Machines," *arXiv:1703.02136 [cs]*, 2017.

[21] E. W. Healy, S. E. Yoho, J. Chen, Y. Wang, and D. Wang, "An algorithm to increase speech intelligibility for hearing-impaired listeners in novel segments of the same noise type," *J. Acoust. Soc. Am.*, vol. 138, no. 3, pp. 1660–1669, 2015.

[22] T. Goehring, F. Bolner, J. J. M. Monaghan, B. van Dijk, A. Zarowski, and S. Bleeck, "Speech enhancement based on neural networks improves speech intelligibility in noise for cochlear implant users," *Hearing Research*, vol. 344, pp. 183–194, 2017.

[23] H. Erdogan, J. R. Hershey, S. Watanabe, and J. L. Roux, "Deep Recurrent Networks for Separation and Recognition of Single Channel Speech in Nonstationary Background Audio," in *New Era for Robust Speech Recognition: Exploiting Deep Learning*. Springer, 2017.

[24] M. Kolbæk, D. Yu, Z.-H. Tan, and J. Jensen, "Multi-talker Speech Separation with Utterance-level Permutation Invariant Training of Deep Recurrent Neural Networks," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 25, no. 10, pp. 1901–1913, 2017.

[25] Y. Wang, A. Narayanan, and D. Wang, "On Training Targets for Supervised Speech Separation," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 22, no. 12, pp. 1849–1858, 2014.

[26] J. Garofolo, D. Graff, P. Doug, and D. Pallett, "CSR-I (WSJ0) Complete LDC93s6a," 1993, philadelphia: Linguistic Data Consortium.

[27] "ITU Rec. P.56 : Objective measurement of active speech level," 1993, https://www.itu.int/rec/T-REC-P.56/.

[28] J. S. Garofolo, L. F. Lamel, W. M. Fisher, J. G. Fiscus, D. S. Pallett, and N. L. Dahlgren, "DARPA TIMIT Acoustic Phonetic Continuous Speech Corpus CDROM," 1993.

[29] J. Barker, M. Ricard, V. Vincent, and S. Watanabe, "The third 'CHiME' Speech Separation and Recognition Challenge: Dataset, task and baselines," in *Proc. ASRU*, 2015.

[30] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput*, vol. 9, no. 8, pp. 1735–1780, 1997.

[31] A. Agarwal *et al.*, "An introduction to computational networks and the computational network toolkit," Microsoft Technical Report {MSR-TR}-2014-112, Tech. Rep., 2014.

[32] E. Vincent, R. Gribonval, and C. Fevotte, "Performance measurement in blind audio source separation," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 14, no. 4, pp. 1462–1469, 2006.

[33] J. Jensen and C. H. Taal, "An Algorithm for Predicting the Intelligibility of Speech Masked by Modulated Noise Maskers," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 24, no. 11, pp. 2009–2022, 2016.

[34] P. C. Loizou, *Speech Enhancement: Theory and Practice*. CRC Press, 2013.

[35] J. S. Erkelens, R. C. Hendriks, R. Heusdens, and J. Jensen, "Minimum Mean-Square Error Estimation of Discrete Fourier Coefficients With Generalized Gamma Priors," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 15, no. 6, pp. 1741–1752, 2007.

Paper F

Monaural Speech Enhancement Using Deep Neural Networks by Maximizing a Short-Time Objective Intelligibility Measure

Morten Kolbæk, Zheng-Hua Tan, and Jesper Jensen

The paper has been published in
Proceedings IEEE International Conference on Acoustics, Speech, and Signal Processing, pp. 5059-5063, April 2018.

© 2018 IEEE
The layout has been revised.

Abstract

In this paper we propose a Deep Neural Network (DNN) based Speech Enhancement (SE) system that is designed to maximize an approximation of the Short-Time Objective Intelligibility (STOI) measure. We formalize an approximate-STOI cost function and derive analytical expressions for the gradients required for DNN training and show that these gradients have desirable properties when used together with gradient based optimization techniques.

We show through simulation experiments that the proposed SE system achieves large improvements in estimated speech intelligibility, when tested on matched and unmatched natural noise types, at multiple signal-to-noise ratios. Furthermore, we show that the SE system, when trained using an approximate-STOI cost function performs on par with a system trained with a mean squared error cost applied to short-time temporal envelopes. Finally, we show that the proposed SE system performs on par with a traditional DNN based Short-Time Spectral Amplitude (STSA) SE system in terms of estimated speech intelligibility. These results are important because they suggest that traditional DNN based STSA SE systems might be optimal in terms of estimated speech intelligibility.

1 Introduction

Design and development of Speech Enhancement (SE) algorithms capable of improving speech quality and intelligibility has been a long-lasting goal in both academia and industry [1, 2]. Such algorithms are useful for a wide range of applications e.g. for mobile communications devices and hearing assistive devices [1].

Despite a large research effort for more than 30 years [1–3] modern single-microphone SE algorithms still perform unsatisfactorily in the complex acoustic environments, which users of e.g. hearing assistive devices are exposed to on a daily basis, e.g. traffic noise, cafeteria noise, or competing speakers.

Traditionally, SE algorithms have been divided into at least two groups; statistical-model based techniques and data-driven techniques. The first group encompasses techniques such as spectral subtraction, the Wiener filter and the short-time spectral amplitude minimum mean squared error estimator [1–3]. These techniques make statistical assumptions about the probability distributions of the speech and noise signals, that enable them to suppress the noise dominated time-frequency regions of the noisy speech signal. In particularly, for stationary noise types this type of algorithms may perform well in terms of speech quality, but in general these techniques do not improve speech intelligibility [4–6]. The second group encompasses data-driven or machine learning techniques e.g. based on non-negative matrix factorization [7], support vector machines [8], and Deep Neural Networks (DNNs) [9, 10].

These techniques make no statistical assumptions. Instead, they learn to suppress noise by observing a large number of representative pairs of noisy and noise-free speech signals in a supervised learning process. SE algorithms based on DNNs can, to some extent, improve speech intelligibility for hearing impaired and normal hearing people, in noisy conditions, if sufficient *a priori* knowledge is available e.g. the identity of the speaker or the noise type. [11–13].

Although the techniques mentioned above are fundamentally different, they typically share at least two common properties. First, they often aim to minimize a Mean Squared Error (MSE) cost function, and secondly, they operate on short frames ($\approx 20 - 30$ ms) in the Short-Time discrete Fourier Transform (STFT) domain [1, 2]. However, it is well known [2, 14] that the human auditory system has a non-linear frequency sensitivity, which is often approximated using e.g. a Gammatone or a one-third octave filter bank [2]. Furthermore, it is known that preservation of modulation frequencies below 7 Hz is critical for speech intelligibility [14, 15]. This suggests that SE algorithms aimed at the human auditory system could benefit by incorporating such information. Numerous works exist, e.g. [10, 16–23] and [1, Sec. 2.2.3] and the references therein, where SE algorithms have been designed with perceptual aspects in mind. However, although these algorithms do take some perceptual aspects into account, they do not directly optimize for speech intelligibility.

In this paper we propose an SE system that maximizes an objective speech intelligibility estimator. Specifically, we design a DNN based SE system that maximizes an approximation of the Short-Time Objective Intelligibility (STOI) [24] measure. The STOI measure has been found to be highly correlated with intelligibility as measured in human listening tests [2, 24]. We derive analytical expressions for the required gradients used for the DNN weight updates during training and use these closed-form expressions to identify desirable properties of the approximate-STOI cost function. Finally, we study the potential performance gain between the proposed approximate-STOI cost function with a classical MSE cost function. We note that our goal is not to achieve state-of-the-art STOI improvements per se, but rather to study and compare the proposed approximate-STOI based SE system to existing DNN based enhancement schemes. Further improvement may straightforwardly be achieved with larger datasets and complex models like long short-term memory recurrent, or convolutional, neural networks [25].

2 Speech Enhancement System

In the following we introduce the approximate-STOI measure and we present the DNN framework used to maximize it. Finally, we discuss techniques used

to reconstruct the enhanced and approximate-STOI optimal speech signal in the time-domain.

2.1 Approximating Short-Time Objective Intelligibility

Let $x[n]$ be the n^{th} sample of the clean time-domain speech signal and let a noisy observation $y[n]$ be defined as

$$y[n] = x[n] + z[n], \tag{F.1}$$

where $z[n]$ is an additive noise sample. Furthermore, let $x(k,m)$ and $y(k,m)$, $k = 1, 2, \ldots, \frac{K}{2} + 1$, $m = 1, 2, \ldots M$, be the single-sided magnitude spectra of the K-point Short-Time discrete Fourier Transforms (STFT) of $x[n]$ and $y[n]$, respectively, where M is the number of STFT frames. Also, let $\hat{x}(k,m)$ be an estimate of $x(k,m)$ obtained as $\hat{x}(k,m) = \hat{g}(k,m)y(k,m)$ where $\hat{g}(k,m)$ is an estimated gain value. In this study we use a 10 kHz sample frequency and a 256 point STFT, i.e. $K = 256$, with a Hann-window size of 256 samples (25.6 ms) and a 128 sample frame shift (12.8 ms). Similarly to STOI [24], we define a short-time temporal envelope vector of the j^{th} one-third octave band for the clean speech signal as

$$\mathbf{x}_{j,m} = [X_j(m-N+1),\ X_j(m-N+2), \ldots, X_j(m)]^T, \tag{F.2}$$

where

$$X_j(m) = \sqrt{\sum_{k=k_1(j)}^{k_2(j)} x(k,m)^2}, \tag{F.3}$$

and $k_1(j)$ and $k_2(j)$ denote the first and last STFT bin index of the j^{th} one-third octave band, respectively. Similarly, we define $\mathbf{y}_{j,m}$ and $Y_j(m)$ for the noisy observation. Also, let $\hat{\mathbf{x}}_{j,m} = diag(\hat{\mathbf{g}}_{j,m})\mathbf{y}_{j,m}$ be the short-time temporal one-third octave band envelope vector of the enhanced speech signal, where $\hat{\mathbf{g}}_{j,m}$ is a gain vector defined in the j^{th} one-third octave band and $diag(\hat{\mathbf{g}}_{j,m})$ is a diagonal matrix with the elements of $\hat{\mathbf{g}}_{j,m}$ on the main diagonal. We use $N = 30$ such that the short-time temporal one-third octave band envelope vectors will span a duration of 384 ms, which ensures that important modulation frequencies are captured [24]. In total, $J = 15$ one-third octave bands are used with the first band having a center frequency of 150 Hz and the last one of approximately 3.8 kHz. These frequencies are chosen such that they span the frequency range in which human speech normally lie [24]. For mathematical tractability, we discard the clipping step[1], otherwise performed by STOI [24],

[1] It has been observed empirically, that omitting the clipping step most often does not affect the performance of STOI, e.g. [20, 26–28].

and define the approximated STOI measure as

$$\mathcal{L}(\mathbf{x}_{j,m}, \hat{\mathbf{x}}_{j,m}) = \frac{\left(\mathbf{x}_{j,m} - \mu_{\mathbf{x}_{j,m}}\right)^T \left(\hat{\mathbf{x}}_{j,m} - \mu_{\hat{\mathbf{x}}_{j,m}}\right)}{\left\|\mathbf{x}_{j,m} - \mu_{\mathbf{x}_{j,m}}\right\| \left\|\hat{\mathbf{x}}_{j,m} - \mu_{\hat{\mathbf{x}}_{j,m}}\right\|}, \tag{F.4}$$

where $\|\cdot\|$ is the euclidean ℓ^2-norm and $\mu_{\mathbf{x}_{j,m}}$ and $\mu_{\hat{\mathbf{x}}_{j,m}}$ are the sample means of $\mathbf{x}_{j,m}$ and $\hat{\mathbf{x}}_{j,m}$, respectively. Obviously, $\mathcal{L}(\mathbf{x}_{j,m}, \hat{\mathbf{x}}_{j,m})$ is simply the Envelope Linear Correlation (ELC) between the vectors $\mathbf{x}_{j,m}$ and $\hat{\mathbf{x}}_{j,m}$.

2.2 Maximizing Approximated STOI Using DNNs

The approximated STOI measure given by Eq. (F.4) is defined in a one-third octave band domain and our goal is to find $\hat{\mathbf{x}}_{j,m} = diag(\hat{\mathbf{g}}_{j,m})\mathbf{y}_{j,m}$ such that Eq. (F.4) is maximized, i.e. finding an optimal gain vector $\hat{\mathbf{g}}_{j,m}$. In this study we estimate these optimal gains using DNNs. Specifically, we use Eq. (F.4) as a cost function and train multiple feed-forward DNNs, one for each one-third octave band, to estimate gain vectors $\hat{\mathbf{g}}_{j,m}$, such that the approximated STOI measure is maximized. For the remainder of this paragraph we omit the subscripts j and m for convenience.

Most modern deep learning toolkits, e.g. Microsoft Cognitive Toolkit (CNTK) [29], perform automatic differentiation, which allow one to train a DNN with a custom cost function, without the need of computing the gradients of the cost function explicitly [25]. Nevertheless, when working with cost functions that have not yet been exhaustively studied, such as the approximated STOI measure, an analytic expression of the gradient can be valuable for studying important properties, such as gradient ℓ^2-norm. It can be shown (details omitted due to space limitations) that the gradient of Eq. (F.4), with respect to the desired signal vector $\hat{\mathbf{x}}$, is given by

$$\nabla\mathcal{L}(\mathbf{x}, \hat{\mathbf{x}}) = \left[\frac{\partial\mathcal{L}(\mathbf{x}, \hat{\mathbf{x}})}{\partial\hat{x}_1}, \frac{\partial\mathcal{L}(\mathbf{x}, \hat{\mathbf{x}})}{\partial\hat{x}_2}, \ldots, \frac{\partial\mathcal{L}(\mathbf{x}, \hat{\mathbf{x}})}{\partial\hat{x}_N}\right]^T, \tag{F.5}$$

where

$$\frac{\partial\mathcal{L}(\mathbf{x}, \hat{\mathbf{x}})}{\partial\hat{x}_m} = \frac{\mathcal{L}(\mathbf{x}, \hat{\mathbf{x}})\,(x_m - \mu_{\mathbf{x}})}{(\hat{\mathbf{x}} - \mu_{\hat{\mathbf{x}}})^T (\mathbf{x} - \mu_{\mathbf{x}})} - \frac{\mathcal{L}(\mathbf{x}, \hat{\mathbf{x}})\,(\hat{x}_m - \mu_{\hat{\mathbf{x}}})}{(\hat{\mathbf{x}} - \mu_{\hat{\mathbf{x}}})^T (\hat{\mathbf{x}} - \mu_{\hat{\mathbf{x}}})}, \tag{F.6}$$

is the partial derivative of $\mathcal{L}(\mathbf{x}, \hat{\mathbf{x}})$ with respect to entry m of $\hat{\mathbf{x}}$.

Furthermore, it can be shown that the ℓ^2-norm of the gradient as formulated by Eqs. (F.5) and (F.6), is given by

$$\|\nabla\mathcal{L}(\mathbf{x}, \hat{\mathbf{x}})\| = \sqrt{1 - \mathcal{L}(\mathbf{x}, \hat{\mathbf{x}})^2}\,\|\hat{\mathbf{x}}\|^{-1}, \tag{F.7}$$

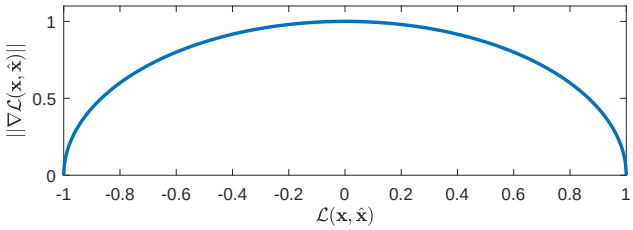


Fig. F.1: ℓ^2-norm of Eq. (F.5) as function of cost function value.

which is shown in Fig. F.1 as function of $\mathcal{L}(\mathbf{x},\hat{\mathbf{x}})$ for the complete range $[-1,1]$, and for $\|\hat{\mathbf{x}}\| = 1$. We see from Fig. F.1 that the ℓ^2-norm of $\mathcal{L}(\mathbf{x},\hat{\mathbf{x}})$ is a concave function with a global maximum at $\mathcal{L}(\mathbf{x},\hat{\mathbf{x}}) = 0$ and is symmetric around zero. We also observe that $\|\nabla\mathcal{L}(\mathbf{x},\hat{\mathbf{x}})\|$ is monotonically decreasing when $\mathcal{L}(\mathbf{x},\hat{\mathbf{x}}) < 0$ and $\mathcal{L}(\mathbf{x},\hat{\mathbf{x}}) > 0$ with $\|\nabla\mathcal{L}(\mathbf{x},\hat{\mathbf{x}})\| = 0$ when $\mathbf{x}$ and $\hat{\mathbf{x}}$ are either perfectly correlated or perfectly anti-correlated. Since $\|\nabla\mathcal{L}(\mathbf{x},\hat{\mathbf{x}})\|$ is large when $\mathbf{x}$ and $\hat{\mathbf{x}}$ are uncorrelated and zero when perfectly correlated, and $\|\nabla\mathcal{L}(\mathbf{x},\hat{\mathbf{x}})\| \neq 0$ otherwise, Eq. (F.4) is well suited as a cost function for gradient-based optimization techniques, such as Stochastic Gradient Descent (SGD) [25], since it guarantees non-zero step lengths for all inputs during optimization except at the optimal solution. In practice, to apply SGD we minimize $-\mathcal{L}(\mathbf{x},\hat{\mathbf{x}})$.

2.3 Reconstructing Approximate-STOI Optimal Speech

When a gain vector $\hat{\mathbf{g}}_{j,m}$ has been estimated by a DNN, the enhanced speech envelope in the one-third octave band domain can be computed as $\hat{\mathbf{x}}_{j,m} = diag(\hat{\mathbf{g}}_{j,m})\mathbf{y}_{j,m}$. However, what we are really interested in is $\hat{x}(k,m)$, i.e. the estimated speech signal in the STFT domain, since $\hat{x}(k,m)$ can straightforwardly be transformed into the time-domain using the overlap-and-add technique[2]. We therefore seek a mapping from the gain vector $\hat{\mathbf{g}}_{j,m}$ estimated in the one-third octave band domain, to the gain $\hat{g}(k,m)$, for a single STFT coefficient. To do so, let $\hat{g}_j(m)$ denote the gain value estimated by a DNN to be applied to the noisy one-third octave band amplitude in frame m. We can then derive the relationship between the gain value $\hat{g}_j(m) \geq 0$ in the one-third octave band, and the corresponding gain values $\hat{g}(k,m) \geq 0$ in the STFT domain as

$$\hat{X}_j(m) = \hat{g}_j(m)Y_j(m) = \sqrt{\sum_{k=k_1(j)}^{k_2(j)-1} \left(\hat{g}(k,m)y(k,m)\right)^2}. \tag{F.8}$$

One solution to Eq (F.8) is

$$\hat{g}_j(m) = \hat{g}(k,m),\ k = k_1(j),\ldots k_2(j)-1. \tag{F.9}$$

Generally, the solution in Eq. (F.9) is not unique; many choices of $\hat{g}(k,m)$ exist that give rise to the same estimated one-third octave band $\hat{X}_j(m)$ (and hence the same value of $\mathcal{L}(\mathbf{x},\hat{\mathbf{x}})$). We choose, for convenience, a uniform gain across the STFT coefficients within a one-third octave band. Since envelope estimates $\hat{X}_j(m)$ are computed for successive values of m, N estimates exist for each $\hat{X}_j(m)$, which are averaged during enhancement. When reconstructing the enhanced speech signal in the time domain, we use the overlap-and-add technique using the phase of the noisy STFT coefficients [2].

3 Experimental Design

To evaluate the performance of the approximate-STOI optimal DNN based SE system we have conducted series of experiments involving multiple matched and unmatched noise types at various SNRs.

3.1 Noisy Speech Mixtures

The clean speech signals used for training all models are from the Wall Street Journal (WSJ0) corpus [30]. The utterances used for training and validation are generated by randomly selecting utterances from 44 male and 47 female speakers from the WSJ0 training set entitled si_tr_s. The training and validation sets consist of 20000 and 2000 utterances, respectively, which is equivalent to approximately 37 hours of training data and 4 hours of validation data. The test set is similarly generated using utterances from 16 speakers from the WSJ0 validation set si_dt_05 and evaluation set si_et_05, and consists of 1000 mixtures or approximately 2 hours of data, see [31] for further details. Notice, the speakers in the test set are different from the speakers in the validation and training sets.

We use six different noise types: two synthetic signals and four noise signals recorded in real-life. The synthetic noise signals encompass a stationary Speech Shaped Noise (SSN) signal and a highly non-stationary 6-speaker Babble (BBL) noise. For real-life noise signals we use the street (STR), cafeteria (CAF), bus (BUS), and pedestrian (PED) noise signals from the CHiME3 dataset [32]. The SSN noise signal is Gaussian white noise, shaped according to the long-term spectrum of the TIMIT corpus [33]. Similarly, the BBL noise signal is constructed by mixing utterances from TIMIT. Further details on the design of the SSN and BBL noise signals can be found in [13]. All noise signals are split into non-overlapping sequences with a 40 min. training sequence, a 5 min. validation sequence and a 5 min. test sequence, i.e. there is no overlap between the noise sequences used for training, validation and test.

Table F.1: Training conditions for different SE systems.

ID:	S0	S1	S2	S3	S4	S5	S6	S7	S8	S9
Cost:	ELC	ELC	ELC	ELC	ELC	EMSE	EMSE	EMSE	EMSE	EMSE
Noise:	SSN	BBL	CAF	STR	ALL	SSN	BBL	CAF	STR	ALL

The noisy speech signals used for training and testing are constructed using Eq. (F.1), where a clean speech signal $x[n]$ is added to a noise sequence $z[n]$ of equal length. To achieve a certain SNR, the noise signal is scaled based on the active speech level of the clean speech signal as per ITU P.56 [34]. The SNRs used for the training and validation sets are chosen uniformly from $[-5, 10]$ dB. The SNR range is chosen to ensure that SNRs are included where intelligibility ranges from degraded to perfectly intelligible.

3.2 Model Architecture and Training

To evaluate the performance of the proposed SE system a total of ten systems, identified as S0 – S9, have been trained using different cost functions and noise types as presented in Table F.1. Five systems (S0–S4) have been trained using the ELC loss from Eq. (F.4) and five systems (S5–S9) have been trained using a standard MSE loss, denoted as Envelope Mean Squared Error (EMSE), since it operates on short-time temporal one-third octave band envelope vectors. This is to investigate the potential performance difference between models trained with an approximate-STOI loss and models trained with the commonly used MSE loss. Eight systems (S0–S3 and S5–S8) are trained as noise type specific systems, i.e. they are trained using only a single noise type. Two systems (S4 and S9) are trained as noise type general systems, i.e. they are trained on all noise types (Noise: "ALL" in Table F.1). This is to investigate the performance drop, if any, when a single system is trained to handle multiple noise types.

Each DNN consists of three hidden layers with 512 units with ReLU activation functions and a sigmoid output layer. The DNNs are trained using SGD with the backpropagation technique and batch normalization [25]. The DNNs are trained for a maximum of 200 epochs with a minibatch size of 256 randomly selected short-time temporal one-third octave band envelope vectors and the learning rates were set to 0.01, and $5 \cdot 10^{-5}$ per sample initially, for S0–S4, and S5–S9, respectively. The learning rates were scaled down by 0.7 when the training cost increased on the validation set. The training was terminated when the learning rate was below 10^{-10}. The different learning rates for the systems trained with the ELC cost function and the systems trained with the EMSE cost functions were found from preliminary experiments. All models were implemented using CNTK [29] and the script files needed to reproduce the reported results can be found in [31].

Table F.2: ELC results for S0 – S9 tested with SSN, BBL, CAF, and STR

	SSN					BBL				
SNR [dB]	UP.	S0 (ELC)	S5 (EMSE)	S4 (ELC)	S9 (EMSE)	UP.	S1 (ELC)	S6 (EMSE)	S4 (ELC)	S9 (EMSE)
-5	0.36	0.66	0.65	0.64	0.63	0.34	0.50	0.51	0.48	0.48
0	0.52	0.77	0.76	0.75	0.74	0.50	0.69	0.69	0.67	0.67
5	0.66	0.82	0.81	0.80	0.79	0.64	0.78	0.77	0.77	0.77
Avg.	0.51	0.75	0.74	0.73	0.72	0.49	0.66	0.66	0.64	0.64

	CAF					STR				
SNR [dB]	UP.	S2 (ELC)	S7 (EMSE)	S4 (ELC)	S9 (EMSE)	UP.	S3 (ELC)	S8 (EMSE)	S4 (ELC)	S9 (EMSE)
-5	0.43	0.61	0.59	0.58	0.58	0.45	0.70	0.68	0.68	0.66
0	0.57	0.73	0.71	0.72	0.70	0.58	0.78	0.76	0.77	0.75
5	0.68	0.79	0.78	0.79	0.77	0.69	0.82	0.80	0.81	0.79
Avg.	0.56	0.71	0.69	0.70	0.68	0.57	0.77	0.75	0.75	0.73

4 Experimental Results

We have evaluated the performance of the ten systems based on their average ELC and STOI scores computed on the test set. The STOI score is computed using the enhanced and reconstructed time-domain speech signal, whereas the ELC score is computed using short-time one-third octave band temporal envelope vectors.

4.1 Matched and Unmatched Noise Type Experiments

In Table F.2 we compare the ELC scores for the noise type specific systems trained using the ELC (S0–S4), and EMSE (S5–S8) cost functions, and tested in matched noise-type conditions (SSN, BBL, CAF, and STR) at an input SNR of -5, 0, and 5 dB. Results covering the SNR range from -10 to 20 dB can be found in [31]. All models achieve large improvements in ELC with an average improvement of approximately 0.15-0.20, for all SNRs and noise types, compared to the ELC score of the noisy, unprocessed signals (denoted UP. in Tables F.2 to F.4). We also see that, as expected, models trained with the ELC cost function (S0–S4) in general achieve similar or slightly higher ELC scores compared to the models trained with EMSE (S5–S8). In Table F.3 we report the STOI scores for the systems in Table F.2 tested in identical conditions. We see moderate to large improvements in STOI in all conditions with an average improvement from 0.07–0.13. We also observe that the systems trained with the EMSE cost function achieve similar improvement in STOI as the systems trained with the ELC cost function. In Table F.4, the ELC and STOI scores for

Table F.3: STOI results for S0 – S9 tested with SSN, BBL, CAF, and STR

	SSN					BBL				
SNR [dB]	UP.	S0 (ELC)	S5 (EMSE)	S4 (ELC)	S9 (EMSE)	UP.	S1 (ELC)	S6 (EMSE)	S4 (ELC)	S9 (EMSE)
-5	0.61	0.78	0.78	0.76	0.76	0.59	0.66	0.67	0.65	0.65
0	0.74	0.88	0.88	0.87	0.87	0.72	0.82	0.82	0.81	0.81
5	0.85	0.93	0.93	0.92	0.92	0.83	0.90	0.90	0.89	0.90
Avg.	0.73	0.86	0.86	0.85	0.85	0.71	0.79	0.80	0.78	0.79

	CAF					STR				
SNR [dB]	UP.	S2 (ELC)	S7 (EMSE)	S4 (ELC)	S9 (EMSE)	UP.	S3 (ELC)	S8 (EMSE)	S4 (ELC)	S9 (EMSE)
-5	0.67	0.76	0.76	0.75	0.75	0.68	0.81	0.82	0.80	0.80
0	0.78	0.86	0.86	0.85	0.86	0.78	0.88	0.89	0.88	0.88
5	0.87	0.91	0.92	0.91	0.92	0.87	0.92	0.93	0.92	0.92
Avg.	0.77	0.84	0.85	0.84	0.84	0.78	0.87	0.88	0.87	0.87

the noise type general systems (S4 and S9) tested with the unmatched BUS and PED noise types are summarized. We see average improvement in the order of 0.1–0.18 in terms of ELC score and 0.05 – 0.09 in terms of STOI. We also see the performance gap between the S4 system (trained with ELC cost function) is small compared to the S9 system (trained with EMSE cost function) and that noise specific systems perform slightly better than the noise general one. The results in Tables F.2 to F.4 are interesting since they show roughly identical global behavior as measured by ELC and STOI for systems trained with the ELC and EMSE cost functions.

4.2 Gain Similarities Between ELC and EMSE Based Systems

We now study to which extent ELC and EMSE based systems behave similarly on a more detailed level. Specifically, we compute correlation coefficients between the gain vectors produced by each of the two types of systems, for SSN, BBL, and STR noise types, and summarize them in Table F.5. In Table F.5 we observe that high sample correlations (> 0.90) are achieved for all noise types and both SNRs, which indicates that the gains produced by a system trained with the ELC cost function are quite similar to the gains produced by a system trained with the EMSE cost function, which supports the findings in Sec. 4.1. Similar conclusions can be drawn for the remaining noise types (results omitted due to space limitations, see [31]).

Table F.4: ELC and STOI for S4 and S9 tested with BUS and PED.

	ELC						STOI					
	BUS			PED			BUS			PED		
SNR	UP.	S4	S9	UP.	S4	S9	UP.	S4	S9	UP.	S4	S9
-5	0.56	0.71	0.68	0.35	0.55	0.53	0.77	0.84	0.84	0.60	0.71	0.71
0	0.66	0.79	0.76	0.50	0.70	0.68	0.85	0.90	0.90	0.72	0.83	0.83
5	0.74	0.83	0.81	0.64	0.78	0.76	0.91	0.94	0.94	0.83	0.90	0.90
Avg.	0.65	0.78	0.75	0.50	0.68	0.66	0.84	0.89	0.89	0.72	0.81	0.81

Table F.5: Sample linear correlation between gain vectors.

SNR [dB]	SSN	BBL	STR
-5	0.93	0.91	0.92
5	0.94	0.96	0.92

Table F.6: STOI score for classical DNN, tested with BBL.

SNR [dB]	UP.	# units	
		512	4096
-5	0.59	0.64	0.66
5	0.83	0.91	0.92

4.3 Approximate-STOI Optimal DNN vs. Classical SE DNN

As a final study we compare the performance of an approximate-STOI optimal DNN based SE system with classical Short-Time Spectral Amplitude (STSA) DNN based enhancement systems that estimate $\hat{g}(k,m)$ directly for each STFT frame (see e.g. [35, 36]). Similarly to S0–S9 these systems are three-layered feed-forward DNNs and use 30 STFT frames as input, but differently from S0–S9, they minimize the MSE between STFT magnitude spectra, i.e. across frequency. The DNNs estimate five STFT frames per time-step and overlapping frames are averaged to construct the final gain. We have trained two of these classical systems, with 512 units and 4096 units, respectively, in each hidden layer, using the BBL noise corrupted training set. The results are presented in Table F.6.

From Table F.6 we see, for example, that such classical STSA-DNN based SE systems trained and tested with BBL noise achieve a maximum STOI score of 0.66 at an input SNR of -5 dB, which is equivalent to the STOI score of 0.66 achieved by S1 in Table F.3. We also see that the classical system performs on par with S1 at an input SNR of 5 dB SNR with a STOI score of 0.92 compared to 0.90 achieved by S1. Although surprising, this is an interesting result since it indicates that no improvement in STOI can be gained by a DNN based SE system that is designed to maximize an approximate-STOI measure using short-time temporal one-third octave band envelope vectors. The important implication of this is that traditional STSA-DNN based SE systems may be close to optimal from an estimated speech intelligibility perspective.

5 Conclusion

In this paper we proposed a Speech Enhancement (SE) system based on Deep Neural Networks (DNNs) that optimizes an approximation of the Short-Time Objective Intelligibility (STOI) estimator. We proposed an approximate-STOI cost function and derived closed-form expressions for the required gradients. We showed that DNNs designed to maximize approximate-STOI, achieve large improvement in STOI when tested in matched and unmatched noise types at various SNRs. We also showed that approximate-STOI optimal systems do not outperform systems that minimize a mean squared error cost. Finally, we showed that approximate-STOI DNN based SE systems perform on par with classical DNN based SE systems. Our findings suggest that a potential speech intelligibility gain of approximate-STOI optimal systems over MSE based systems is modest at best.

References

[1] R. C. Hendriks, T. Gerkmann, and J. Jensen, "DFT-Domain Based Single-Microphone Noise Reduction for Speech Enhancement: A Survey of the State of the Art," *Synth. Lect. on Speech and Audio Process.*, vol. 9, no. 1, pp. 1–80, Jan. 2013.

[2] P. C. Loizou, *Speech Enhancement: Theory and Practice*. CRC Press, 2013.

[3] Y. Ephraim and D. Malah, "Speech enhancement using a minimum-mean square error short-time spectral amplitude estimator," *IEEE Trans. Acoust., Speech, and Sig. Process.*, vol. 32, no. 6, pp. 1109–1121, 1984.

[4] Y. Hu and P. C. Loizou, "A comparative intelligibility study of single-microphone noise reduction algorithms," *J. Acoust. Soc. Am.*, vol. 122, no. 3, pp. 1777–1786, Sep. 2007.

[5] H. Luts *et al.*, "Multicenter evaluation of signal enhancement algorithms for hearing aids," *J. Acoust. Soc. Am.*, vol. 127, no. 3, pp. 1491–1505, 2010.

[6] J. Jensen and R. Hendriks, "Spectral Magnitude Minimum Mean-Square Error Estimation Using Binary and Continuous Gain Functions," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 20, no. 1, pp. 92–102, Jan. 2012.

[7] E. M. Grais and H. Erdogan, "Single channel speech music separation using nonnegative matrix factorization and spectral masks," in *Proc. ICDSP*, 2011, pp. 1–6.

[8] Y. Wang and D. Wang, "Towards Scaling Up Classification-Based Speech Separation," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 21, no. 7, pp. 1381–1390, Jul. 2013.

[9] Y. Xu, J. Du, L.-R. Dai, and C.-H. Lee, "A Regression Approach to Speech Enhancement Based on Deep Neural Networks," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 23, no. 1, pp. 7–19, Jan. 2015.

[10] E. W. Healy, S. E. Yoho, J. Chen, Y. Wang, and D. Wang, "An algorithm to increase speech intelligibility for hearing-impaired listeners in novel segments of the same noise type," *J. Acoust. Soc. Am.*, vol. 138, no. 3, pp. 1660–1669, 2015.

[11] J. Chen, Y. Wang, S. E. Yoho, D. Wang, and E. W. Healy, "Large-scale training to increase speech intelligibility for hearing-impaired listeners in novel noises," *J. Acoust. Soc. Am.*, vol. 139, no. 5, pp. 2604–2612, 2016.

[12] E. W. Healy, M. Delfarah, J. L. Vasko, B. L. Carter, and D. Wang, "An algorithm to increase intelligibility for hearing-impaired listeners in the presence of a competing talker," *J. Acoust. Soc. Am.*, vol. 141, no. 6, pp. 4230–4239, 2017.

[13] M. Kolbæk, Z. H. Tan, and J. Jensen, "Speech Intelligibility Potential of General and Specialized Deep Neural Network Based Speech Enhancement Systems," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 25, no. 1, pp. 153–167, 2017.

[14] B. Moore, *An Introduction to the Psychology of Hearing*, 6th ed. Brill, 2013.

[15] T. M. Elliott and F. E. Theunissen, "The Modulation Transfer Function for Speech Intelligibility," *PLOS Computational Biology*, vol. 5, no. 3, p. e1000302, Mar. 2009.

[16] Y. Hu and P. C. Loizou, "A perceptually motivated approach for speech enhancement," *IEEE Trans. Speech, Audio, Process.*, vol. 11, no. 5, pp. 457–465, 2003.

[17] Y. Ephraim and D. Malah, "Speech enhancement using a minimum mean-square error log-spectral amplitude estimator," *IEEE Trans. Acoust., Speech, and Sig. Process.*, vol. 33, no. 2, pp. 443–445, 1985.

[18] N. Virag, "Single channel speech enhancement based on masking properties of the human auditory system," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 7, no. 2, pp. 126–137, 1999.

[19] P. C. Loizou, "Speech Enhancement Based on Perceptually Motivated Bayesian Estimators of the Magnitude Spectrum," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 13, no. 5, pp. 857–869, 2005.

[20] L. Lightburn and M. Brookes, "SOBM - a binary mask for noisy speech that optimises an objective intelligibility metric," in *Proc. ICASSP*, 2015, pp. 5078–5082.

[21] W. Han, X. Zhang, G. Min, X. Zhou, and W. Zhang, "Perceptual weighting deep neural networks for single-channel speech enhancement," in *Proc. (WCICA*, 2016, pp. 446–450.

[22] P. G. Shivakumar and P. Georgiou, "Perception Optimized Deep Denoising AutoEncoders for Speech Enhancement - Semantic Scholar," in *INTERSPEECH*, 2016, pp. 3743–3747.

[23] Y. Koizumi, K. Niwa, Y. Hioka, K. Kobayashi, and Y. Haneda, "DNN-based source enhancement self-optimized by reinforcement learning using sound quality measurements," in *Proc. ICASSP*, 2017, pp. 81–85.

[24] C. H. Taal, R. C. Hendriks, R. Heusdens, and J. Jensen, "An Algorithm for Intelligibility Prediction of Time-Frequency Weighted Noisy Speech," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 19, no. 7, pp. 2125–2136, 2011.

[25] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016.

[26] J. Jensen and C. H. Taal, "An Algorithm for Predicting the Intelligibility of Speech Masked by Modulated Noise Maskers," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 24, no. 11, pp. 2009–2022, 2016.

[27] A. H. Andersen, J. M. d. Haan, Z. H. Tan, and J. Jensen, "Predicting the Intelligibility of Noisy and Nonlinearly Processed Binaural Speech," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 24, no. 11, pp. 1908–1920, 2016.

[28] C. H. Taal, R. C. Hendriks, and R. Heusdens, "Matching pursuit for channel selection in cochlear implants based on an intelligibility metric," in *Proc. EUSIPCO*, 2012, pp. 504–508.

[29] A. Agarwal *et al.*, "An introduction to computational networks and the computational network toolkit," Microsoft Technical Report {MSR-TR}-2014-112, Tech. Rep., 2014.

[30] J. Garofolo, D. Graff, P. Doug, and D. Pallett, "CSR-I (WSJ0) Complete LDC93s6a," 1993, philadelphia: Linguistic Data Consortium.

[31] M. Kolbæk, Z.-H. Tan, and J. Jensen, "Supplemental Material." [Online]. Available: http://kom.aau.dk/~mok/icassp2018

[32] J. Barker, M. Ricard, V. Vincent, and S. Watanabe, "The third 'CHiME' Speech Separation and Recognition Challenge: Dataset, task and baselines," in *Proc. ASRU*, 2015.

[33] J. S. Garofolo, L. F. Lamel, W. M. Fisher, J. G. Fiscus, D. S. Pallett, and N. L. Dahlgren, "DARPA TIMIT Acoustic Phonetic Continuous Speech Corpus CDROM," 1993.

[34] ITU, "Rec. P.56 : Objective measurement of active speech level," 1993, https://www.itu.int/rec/T-REC-P.56/.

[35] F. Weninger, J. R. Hershey, J. L. Roux, and B. Schuller, "Discriminatively trained recurrent neural networks for single-channel speech separation," in *GlobalSIP*, 2014, pp. 577–581.

[36] M. Kolbæk, Z. H. Tan, and J. Jensen, "Speech Enhancement using Long Short-Term Memory based Recurrent Neural Networks for Noise Robust Speaker Verification," in *Proc. SLT*, 2016, pp. 305–311.

This page intentionally left blank.

Paper G

On the Relationship between Short-Time Objective Intelligibility and Short-Time Spectral-Amplitude Mean Squared Error for Speech Enhancement

Morten Kolbæk, Zheng-Hua Tan, and Jesper Jensen

The paper is under major revision in
IEEE/ACM Transactions on Audio Speech and Language Processing,
August 2018.

© 2018 IEEE
The layout has been revised.

Abstract

The majority of Deep Neural Network (DNN) based speech enhancement algorithms rely on the Mean Squared Error (MSE) criterion of Short-Time Spectral Amplitudes (STSA), which has no apparent link to human perception, e.g. speech intelligibility. Short-Time Objective Intelligibility (STOI), a popular state-of-the-art speech intelligibility estimator, on the other hand, relies on linear correlation of speech temporal envelopes. This raises the question if a DNN training criterion based on Envelope Linear Correlation (ELC) can lead to improved speech intelligibility performance of DNN based speech enhancement algorithms compared to algorithms based on the STSA-MSE criterion. In this paper we derive that, under certain general conditions, the STSA-MSE and ELC criteria are practically equivalent, and we provide empirical data to support our theoretical results. The important implication of our findings is that the standard STSA minimum-MSE estimator is near optimal, if the objective is to enhance noisy speech in a manner which is optimal with respect to a state-of-the-art speech intelligibility estimator.

1 Introduction

Despite the recent success of Deep Neural Network (DNN) based speech enhancement algorithms [1–5], it is yet unknown if these algorithms are optimal in terms of aspects related to human auditory perception, e.g. speech intelligibility, since existing algorithms do not directly optimize criteria designed with human auditory perception in mind.

Many current state-of-the-art DNN based speech enhancement algorithms use a Mean Squared Error (MSE) training criterion [6–8] on Short-Time Spectral Amplitudes (STSA). This, however, might not be the optimal training criterion if the target is the human auditory system, and improvement in speech intelligibility or speech quality is the desired objective.

It is well known that the frequency sensitivity of the human auditory system is non-linear (e.g. [9, 10]) and, as a consequence, is often approximated in digital signal processing algorithms using e.g. a Gammatone filter bank [11] or a one-third octave band filter bank [12]. It is also well known that preservation of modulation frequencies in the range 4-20 Hz are critical for speech intelligibility [9, 13, 14]. Therefore, it is natural to believe that, if prior knowledge about the human auditory system is incorporated into a speech enhancement algorithm, improvements in speech intelligibility or speech quality can be achieved [15].

Indeed, numerous works exist that attempt to incorporate such knowledge (e.g. [16–26] and references therein). In [16] a transform-domain method based on a Gammatone filter bank was used, which incorporates a non-linear frequency resolution mimicking that of the human auditory system. In [17]

different perceptually motivated cost functions were used to derive STSA clean speech spectrum estimators in order to emphasize spectral peak information, account for auditory masking or penalize spectral over-attenuation. In [20, 21] similar goals were pursued, but instead of using classical statistically-based models, DNNs were used. Finally, in [22] a deep reinforcement learning technique was used to reward solutions that achieved a large score in terms of Perceptual Evaluation of Speech Quality (PESQ) [27], a commonly used speech quality estimator.

Although the works in e.g. [16, 17, 21, 22] include knowledge about the human auditory system the techniques are not designed specifically to maximize speech intelligibility. While speech processing methods that improve speech intelligibility would be of vital importance for applications such as mobile communications, or hearing assistive devices, only very little research has been performed to understand if DNN-based speech enhancement systems can help improve speech intelligibility. Very recent work [23–26] has investigated if DNNs trained to maximize a state-of-the-art speech intelligibility estimator are capable of improving speech intelligibility as measured by the estimator [23–25] or human listeners [26]. Specifically, DNNs were trained to maximize the Short-Time Objective Intelligibility (STOI) [12] estimator and were then compared, in terms of STOI, with DNNs trained to minimize the classical STSA-MSE criterion. Surprisingly, although all DNNs improved STOI, the DNNs trained to maximize STOI showed none or only very modest improvements in STOI compared to the DNNs trained with the classical STSA-MSE criterion [23–26].

The STOI speech intelligibility estimator has proven to be able to quite accurately predict the intelligibility of noisy/processed speech in a large range of acoustic scenarios, including speech processed by mobile communication devices [28], ideal time-frequency weighted noisy speech [12], noisy speech enhanced by single-microphone time-frequency weighting-based speech enhancement systems [12, 29, 30], and speech processed by hearing assistive devices such as cochlear implants [31]. STOI has also been shown to be robust to variations in language types, including Danish [12], Dutch [30], and Mandarin [32]. Finally, recent studies e.g. [6, 7] also show a good correspondence between STOI predictions of noisy speech enhanced by DNN-based speech enhancement systems, and speech intelligibility. As a consequence, STOI is currently the, perhaps, most commonly used speech intelligibility estimator for objectively evaluating the performance of speech enhancement systems [6–8, 16]. Therefore, it is natural to believe that gains in speech intelligibility, as estimated by STOI, can be achieved by utilizing an optimality criterion based on STOI as opposed to the classical criterion based on STSA-MSE.

In this paper we study the potential gain in speech intelligibility that can be achieved, if a DNN is designed to perform optimally with respect to the STOI speech intelligibility estimator. We derive that, under certain general

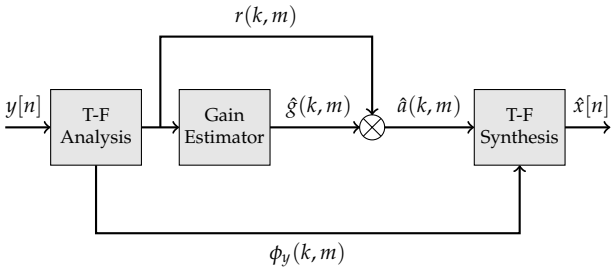


Fig. G.1: Classical gain-based speech enhancement system. The noisy time-domain signal $y[n] = x[n] + v[n]$ is first decomposed into a Time-Frequency (T-F) representation $r(k,m)$ for time-frame m and frequency index k. An estimator, e.g. a DNN, estimates a gain $\hat{g}(k,m)$ that is applied to the noisy short-term magnitude spectrum $r(k,m)$ to arrive at an enhanced signal magnitude $\hat{a}(k,m) = \hat{g}(k,m)r(k,m)$. Finally, the enhanced time-domain signal $\hat{x}[n]$ is obtained from a T-F synthesis stage using the phase of the noisy signal $\phi_y(k,m)$.

conditions, maximizing an approximate-STOI criterion is equivalent to minimizing a STSA-MSE criterion. Furthermore, we present empirical data using simulation studies with DNNs applied to noisy speech signals, that support our theoretical results. Finally, we show theoretically under which conditions the equality between the approximate-STOI criterion and the STSA-MSE criterion holds for practical systems. Our results are in line with recent empirical work and might explain the somewhat surprising result in [23–26], where none or only very modest improvements in STOI were achieved with STOI optimal DNNs compared to STSA-MSE optimal DNNs.

2 STFT-Domain Based Speech Enhancement

Fig. G.1 shows a block-diagram of a classical gain-based speech enhancement system [18, 33]. Let $x[n]$ be the n^{th} sample of the clean time-domain speech signal and let a noisy observation $y[n]$ be given by

$$y[n] = x[n] + v[n], \tag{G.1}$$

where $v[n]$ is a sample of additive noise. Furthermore, let $a(k,m)$ and $r(k,m)$, $k = 1, \ldots, \frac{K}{2} + 1$, $m = 1, \ldots M$, denote the single-sided magnitude spectra of the K-point Short-Time Fourier Transform (STFT) of $x[n]$ and $y[n]$, respectively, where M is the number of STFT frames. Also, let $\hat{a}(k,m)$ denote an estimate of $a(k,m)$ obtained as $\hat{a}(k,m) = \hat{g}(k,m)r(k,m)$. Here, $\hat{g}(k,m)$ is a scalar gain factor applied to the magnitude spectrum of the noisy speech $r(k,m)$ to arrive at an estimate $\hat{a}(k,m)$ of the clean speech magnitude spectrum $a(k,m)$. It is the goal of many STFT-based speech enhancement systems to find appropriate values for $\hat{g}(k,m)$ based on the available noisy signal

$y[n]$. The gain factor $\hat{g}(k,m)$ is typically estimated using either statistical model-based methods such as classical STSA Minimum Mean Squared Error (MMSE) estimators [34], [18, 33], or machine learning based techniques such as Gaussian mixture models [35], support vector machines [36], or, more recently, DNNs [6–8, 16]. For reconstructing the enhanced speech signal in the time domain, it is common practice to append the short-time phase spectrum of the noisy signal to the estimated short-time magnitude spectrum and then use the overlap-and-add technique [37], [33].

3 Short-Time Objective Intelligibility (STOI)

In the following, we shortly review the STOI intelligibility estimator [12]. For further details we refer to [12]. Let the j^{th} one-third octave band clean-speech amplitude, for time-frame m, be defined as

$$a_j(m) = \sqrt{\sum_{k=k_1(j)}^{k_2(j)} a(k,m)^2}, \tag{G.2}$$

where $k_1(j)$ and $k_2(j)$ denote the first and last STFT bin index, respectively, of the j^{th} one-third octave band. Furthermore, let a short-time temporal envelope vector that spans time-frames $m-N+1,\ldots,m$, for the clean speech signal be defined as

$$\underline{a}_{j,m} = [a_j(m-N+1),\ a_j(m-N+2),\ldots,a_j(m)]^T \tag{G.3}$$

In a similar manner we define $\underline{\hat{a}}_{j,m}$ and $\underline{r}_{j,m}$ for the enhanced speech signal and the noisy observation, respectively. The parameter N defines the length of the temporal envelope and for STOI $N=30$[1], which for the STFT settings used in this study, as well as in [12], corresponds to approximately 384 ms. Finally, an approximation[2] of the STOI speech intelligibility estimator for a pair of short-time temporal envelope vectors can then be defined as

$$\mathcal{L}(\underline{a}_{j,m},\underline{\hat{a}}_{j,m}) = \frac{\left(\underline{a}_{j,m}-\mu_{\underline{a}_{j,m}}\right)^T\left(\underline{\hat{a}}_{j,m}-\mu_{\underline{\hat{a}}_{j,m}}\right)}{\left\|\underline{a}_{j,m}-\mu_{\underline{a}_{j,m}}\right\|\ \left\|\underline{\hat{a}}_{j,m}-\mu_{\underline{\hat{a}}_{j,m}}\right\|}, \tag{G.4}$$

where $\|\cdot\|$ denotes the Euclidean ℓ^2-norm and $\mu_{\underline{a}_{j,m}}$ and $\mu_{\underline{\hat{a}}_{j,m}}$ denote the sample means of $\underline{a}_{j,m}$ and $\underline{\hat{a}}_{j,m}$, respectively. Note that $\mathcal{L}(\underline{a}_{j,m},\underline{\hat{a}}_{j,m})$ is nothing more

[1] With $N=30$, STOI is sensitive to temporal modulations of 2.6 Hz and higher, which are frequencies important for speech intelligibility [12].

[2] This is an approximation, since the clipping and normalization steps otherwise used in STOI, have been omitted. This has empirically been found not to have any significant effect on the performance in most cases [19, 29, 38, 39].

than the sample Envelope Linear Correlation (ELC) between the clean and enhanced envelope vectors $\underline{a}_{j,m}$ and $\hat{\underline{a}}_{j,m}$. From $\mathcal{L}(\underline{a}_{j,m}, \hat{\underline{a}}_{j,m})$, the final STOI score for an entire speech signal is then defined as [12] the scalar, $-1 \leq d \leq 1$,

$$d = \frac{1}{J(M-N+1)} \sum_{j=1}^{J} \sum_{m=N}^{M} \mathcal{L}(\underline{a}_{j,m}, \hat{\underline{a}}_{j,m}), \tag{G.5}$$

where J is the number of one-third octave bands and $M - N + 1$ is the total number of short-time temporal envelope vectors. Similarly to [12], we use $J = 15$ with a center frequency of the first one-third octave band at 150 Hz and the last at approximately 3.8 kHz to ensure a frequency range that covers the majority of the spectral information of human speech. The STOI score in general has been shown to often have high correlation with listening tests involving human test subjects, i.e. the higher numerical value of Eq. (G.5), the more intelligible is the speech signal.

Since STOI, as approximated by Eq. (G.5), is a sum of ELC values as given by Eq. (G.4), maximizing Eq. (G.4) will also maximize the overall STOI score in Eq. (G.5). As a consequence, in order to find an estimate $\hat{x}[n]$ of $x[n]$ so that STOI is maximized, one can focus on finding optimal estimates of the individual short-time temporal envelope vectors $\underline{a}_{j,m}$. Therefore, we define $\hat{\underline{a}}_{j,m} = \text{diag}(\hat{\underline{g}}_{j,m})\underline{r}_{j,m}$ as the short-time temporal one-third octave band envelope vector of the enhanced speech signal, where $\hat{\underline{g}}_{j,m}$ is an estimated gain vector and $\text{diag}(\hat{\underline{g}}_{j,m})$ is a diagonal matrix with the elements of $\hat{\underline{g}}_{j,m}$ on the main diagonal.

4 Envelope Linear Correlation Estimator

We now introduce the approximate-STOI criterion in a stochastic context and derive the speech envelope estimator that maximizes it. We denote this estimator as the Maximum Mean Envelope Linear Correlation (MMELC) estimator. Let $A_j(m)$ and $R_j(m)$ denote random variables representing a clean and a noisy, respectively, one-third octave band magnitude, for band j and time frame m. Furthermore, let

$$\underline{A}_j(m) = \left[A_j(m-N+1), \ldots A_j(m)\right], \tag{G.6}$$

and

$$\underline{R}_j(m) = \left[R_j(m-N+1), \ldots R_j(m)\right], \tag{G.7}$$

be the stack of these random variables in random envelope vectors. Finally, in a similar manner, let

$$\hat{\underline{A}}_j(m) = \left[\hat{A}_j(m-N+1), \ldots \hat{A}_j(m)\right], \tag{G.8}$$

be a random envelope vector representing an estimate of $\underline{A}_j(m)$. Now, the contribution of $\hat{\underline{A}}_j(m)$ to speech intelligibility may be approximated as the ELC between the envelope vectors $\underline{A}_j(m)$ and $\hat{\underline{A}}_j(m)$. In the following, the indices j and m are omitted for convenience. Let $\underline{1}$ denote a vector of ones, and let $\underline{\mu}_{\underline{A}} = \frac{1}{N}\underline{1}^T\underline{A}\underline{1}$ be a vector, whose entries equal the sample mean of the entries in $\underline{A}$. Let $\underline{\mu}_{\hat{\underline{A}}}$ be defined in a similar manner. Finally, let the ELC between $\underline{A}$ and $\hat{\underline{A}}$, which is a random variable, be defined as

$$\rho\left(\underline{A}, \hat{\underline{A}}\right) \triangleq \frac{\left(\underline{A} - \underline{\mu}_{\underline{A}}\right)^T \left(\hat{\underline{A}} - \underline{\mu}_{\hat{\underline{A}}}\right)}{\left\|\underline{A} - \underline{\mu}_{\underline{A}}\right\| \left\|\hat{\underline{A}} - \underline{\mu}_{\hat{\underline{A}}}\right\|}, \tag{G.9}$$

and the expected ELC as

$$\begin{aligned} \Omega_{ELC} &= \mathbb{E}_{\underline{A},\underline{R}}\left[\rho\left(\underline{A}, \hat{\underline{A}}\right)\right] \\ &= \int\int \rho\left(\underline{a}, \hat{\underline{a}}\right) f_{\underline{A},\underline{R}}\left(\underline{a}, \underline{r}\right) d\underline{a}\, d\underline{r} \\ &= \int \underbrace{\int \rho\left(\underline{a}, \hat{\underline{a}}\right) f_{\underline{A}|\underline{R}}\left(\underline{a}|\underline{r}\right)\, d\underline{a}}_{\Gamma(\underline{r})}\, f_{\underline{R}}\left(\underline{r}\right)\, d\underline{r}. \end{aligned} \tag{G.10}$$

Here, $\hat{\underline{a}}$ is related to $\underline{r}$ via a deterministic map, e.g. a DNN, and $f_{\underline{A},\underline{R}}\left(\underline{a}, \underline{r}\right)$ denotes the joint Probability Density Function (PDF) of clean and noisy or processed one-third octave band envelope vectors. Furthermore, $f_{\underline{A}|\underline{R}}\left(\underline{a}|\underline{r}\right)$ and $f_{\underline{R}}\left(\underline{r}\right)$ denote a conditional and marginal PDF, respectively.

An optimal estimator can be found by minimizing the Bayes risk [33, 40], which is equivalent to maximizing Eq. (G.10), hence arriving at the MMELC estimator, which we denote as $\hat{\underline{a}}_{MMELC}$. To do so, observe that for a particular noisy observation $\underline{r}$ maximizing $\Gamma\left(\underline{r}\right)$ maximizes Eq. (G.10), since $f_{\underline{R}}\left(\underline{r}\right) \geq 0\ \forall\ \underline{r}$. In other words, our goal is to maximize $\Gamma\left(\underline{r}\right)$ for each and every $\underline{r}$. Hence, for a particular observation, $\underline{r}$, the MMELC estimate is given by

$$\begin{aligned} \hat{\underline{a}}_{MMELC} &= \arg\max_{\hat{\underline{a}}} \int \rho\left(\underline{a}, \hat{\underline{a}}\right) f_{\underline{A}|\underline{R}}\left(\underline{a}|\underline{r}\right)\, d\underline{a} \\ &= \arg\max_{\hat{\underline{a}}} \int \frac{\left(\underline{a} - \underline{\mu}_{\underline{a}}\right)^T \left(\hat{\underline{a}} - \underline{\mu}_{\hat{\underline{a}}}\right)}{\left\|\underline{a} - \underline{\mu}_{\underline{a}}\right\| \left\|\hat{\underline{a}} - \underline{\mu}_{\hat{\underline{a}}}\right\|} f_{\underline{A}|\underline{R}}\left(\underline{a}|\underline{r}\right)\, d\underline{a} \\ &= \arg\max_{\hat{\underline{a}}} \underbrace{\int \frac{\left(\underline{a} - \underline{\mu}_{\underline{a}}\right)^T}{\left\|\underline{a} - \underline{\mu}_{\underline{a}}\right\|} f_{\underline{A}|\underline{R}}\left(\underline{a}|\underline{r}\right)\, d\underline{a}}_{\mathbb{E}_{\underline{A}|\underline{r}}\left[\underline{e}(\underline{A})^T\right]} \underbrace{\frac{\left(\hat{\underline{a}} - \underline{\mu}_{\hat{\underline{a}}}\right)}{\left\|\hat{\underline{a}} - \underline{\mu}_{\hat{\underline{a}}}\right\|}}_{\underline{e}(\hat{\underline{a}})} \\ &= \arg\max_{\hat{\underline{a}}}\ \mathbb{E}_{\underline{A}|\underline{r}}\left[\underline{e}(\underline{A})^T\right] \underline{e}(\hat{\underline{a}}), \end{aligned} \tag{G.11}$$

where $\underline{e}(\cdot)$ is a function that normalizes its vector argument to zero sample mean and unit norm and where we used that for a given noisy observation $\underline{r}$, $\hat{\underline{a}}$ is deterministic. Note that the solution to Eq. (G.11) is non-unique. For one given solution, say $\hat{\underline{a}}^*$, any affine transformation, $\delta\hat{\underline{a}}^* + \gamma\underline{1}\ \forall\ \delta, \gamma \in \mathcal{R}$, is also a solution, because any such transformation is undone by $\underline{e}(\cdot)$. Hence, in the following we focus on finding one such particular solution, namely the zero sample mean, unit norm solution, i.e. the vector $\underline{e}(\hat{\underline{a}})$ that maximizes the inner product with the vector $\mathbb{E}_{\underline{A}|\underline{r}}\left[\underline{e}(\underline{A}|\underline{r})\right]$. To do so, let $\underline{\alpha} = \mathbb{E}_{\underline{A}|\underline{r}}\left[\underline{e}(\underline{A}|\underline{r})\right]$, and let $\underline{e}(\hat{\underline{a}}^*)$ denote the zero sample mean, unit norm vector that maximizes Eq. (G.11). Then, using the method of Lagrange multipliers, it can be shown (see Appendix A) that the MMELC estimator is given by

$$\begin{aligned}\hat{\underline{a}}_{MMELC} &= \underline{e}(\hat{\underline{a}}^*) \\ &= \frac{\left(\underline{\alpha} - \underline{\mu}_{\underline{\alpha}}\right)}{\left\|\underline{\alpha} - \underline{\mu}_{\underline{\alpha}}\right\|} \\ &= \frac{\underline{\alpha}}{\|\underline{\alpha}\|},\end{aligned} \tag{G.12}$$

which is nothing more than the vector $\underline{\alpha}$, normalized to unit norm. The fact that $\underline{\mu}_{\underline{\alpha}} = \frac{1}{N}\underline{1}^T\alpha\underline{1} = \underline{0}$ follows from Eq. (G.11), where it is seen that $\underline{\alpha} = \mathbb{E}_{\underline{A}|\underline{r}}\left[\underline{e}(\underline{A}|\underline{r})\right]$ is an expectation over vectors $(\underline{a} - \underline{\mu}_{\underline{a}})\left\|\underline{a} - \underline{\mu}_{\underline{a}}\right\|^{-1}$ whose sample mean is zero. By interpreting the expectation as an infinite linear combination of such vectors, it follows that $\underline{\mu}_{\underline{\alpha}} = \underline{0}$.

5 Relation to STSA-MMSE Estimators

We now show that the MMELC estimator, Eq. (G.12), is asymptotically equivalent to the one-third octave band STSA-MMSE estimator for large envelope lengths, i.e. as $N \to \infty$. The STSA-MSE (e.g. [34]) is defined as

$$\Omega_{MSE} = \mathbb{E}_{\underline{A},\underline{R}}\left[\left(\underline{A} - \hat{\underline{A}}\right)^2\right]. \tag{G.13}$$

It can be shown (e.g. [18, 33, 34]) that the optimal Bayesian estimator with respect to Eq. (G.13), is the STSA-MMSE estimator given by the conditional mean defined as

$$\begin{aligned}\hat{\underline{a}}_{MMSE} &= \int \underline{a}\ f_{\underline{A}|\underline{R}}\left(\underline{a}|\underline{r}\right)\ d\underline{a} \\ &= \mathbb{E}_{\underline{A}|\underline{r}}\left[\underline{A}|\underline{r}\right].\end{aligned} \tag{G.14}$$

To show that $\hat{\underline{a}}_{MMELC}$ is asymptotically equivalent to $\hat{\underline{a}}_{MMSE}$, let us introduce the idempotent, symmetric matrix

$$\underline{\underline{H}} = \underline{\underline{I}}_N - \frac{1}{N}\underline{1}\underline{1}^T, \tag{G.15}$$

where $\underline{\underline{I}}_N$ denotes the N-dimensional identity matrix. We can then rewrite the vector $\underline{\alpha}$ as

$$\begin{aligned}\underline{\alpha} &= \int \frac{\left(\underline{a} - \underline{\mu}_{\underline{a}}\right)}{\left\|\hat{\underline{a}} - \underline{\mu}_{\hat{\underline{a}}}\right\|} f_{\underline{A}|\underline{R}}\left(\underline{a}|\underline{r}\right)\, d\underline{a} \\ &= \int \frac{\underline{\underline{H}}\underline{a}}{\left\|\underline{\underline{H}}\underline{a}\right\|} f_{\underline{A}|\underline{R}}\left(\underline{a}|\underline{r}\right)\, d\underline{a} \\ &= \mathbb{E}_{\underline{A}|\underline{r}}\left[\frac{\underline{\underline{H}}\underline{A}|\underline{r}}{\left\|\underline{\underline{H}}\underline{A}|\underline{r}\right\|}\right] \\ &= \mathbb{E}_{\underline{A}|\underline{r}}\left[\frac{\underline{Z}}{\left\|\underline{Z}\right\|}\right],\end{aligned} \tag{G.16}$$

where $\underline{A}|\underline{r}$ is a random vector, and we introduced the notation $\underline{Z} \triangleq \underline{\underline{H}}\underline{A}|\underline{r}$. We now employ the following conditional independence assumption

$$f_{\underline{A}|\underline{R}}\left(\underline{a}|\underline{r}\right) = \prod_{j=1}^{N} f_{A_j|R_j=r_j}(a_j|r_j). \tag{G.17}$$

This is a standard assumption in the area of speech enhancement, when operating in the STFT domain and has been the underlying assumption of a very large number of speech enhancement methods (see e.g. [18, 33, 34, 41, 42] and references therein). The conditional independence assumption is, for example, valid, when speech and noise STFT coefficients may be assumed statistically independent across time and frequency and mutually independent [33, 34, 43].

Using Kolmogorovs strong law of large numbers [44, pp. 67-68] and the conditional independence assumption, it can be shown (see Appendix B) that asymptotically, as $N \to \infty$, the expectation in Eq. (G.16) factorizes as

$$\lim_{N\to\infty}\ \underline{\alpha} = \lim_{N\to\infty} \mathbb{E}_{\underline{A}|\underline{r}}\left[\frac{1}{\left\|\underline{Z}\right\|}\right] \mathbb{E}_{\underline{A}|\underline{r}}\left[\underline{Z}\right]. \tag{G.18}$$

Combining this result with Eq. (G.12) leads to

$$\begin{aligned}
\lim_{N\to\infty} \underline{\hat{a}}_{MMELC} &= \lim_{N\to\infty} \frac{\underline{\alpha}}{\left\|\underline{\alpha}\right\|} \\
&= \lim_{N\to\infty} \frac{\mathbb{E}_{\underline{A}|\underline{r}}\left[\frac{1}{\|\underline{Z}\|}\right] \mathbb{E}_{\underline{A}|\underline{r}}\left[\underline{Z}\right]}{\left\|\mathbb{E}_{\underline{A}|\underline{r}}\left[\frac{1}{\|\underline{Z}\|}\right] \mathbb{E}_{\underline{A}|\underline{r}}\left[\underline{Z}\right]\right\|} \\
&= \lim_{N\to\infty} \frac{\mathbb{E}_{\underline{A}|\underline{r}}\left[\frac{1}{\|\underline{Z}\|}\right] \mathbb{E}_{\underline{A}|\underline{r}}\left[\underline{Z}\right]}{\mathbb{E}_{\underline{A}|\underline{r}}\left[\frac{1}{\|\underline{Z}\|}\right] \left\|\mathbb{E}_{\underline{A}|\underline{r}}\left[\underline{Z}\right]\right\|} \\
&= \lim_{N\to\infty} \frac{\mathbb{E}_{\underline{A}|\underline{r}}\left[\underline{Z}\right]}{\left\|\mathbb{E}_{\underline{A}|\underline{r}}\left[\underline{Z}\right]\right\|}.
\end{aligned} \tag{G.19}$$

Since Eq. (G.11) is invariant to affine transformations of its input arguments, we can scale $\underline{\hat{a}}_{MMELC}$ with the scalar quantity $\|\mathbb{E}_{\underline{A}|\underline{r}}\left[\underline{Z}\right]\|$ in Eq. (G.19) to arrive at

$$\lim_{N\to\infty} \ \underline{\hat{a}}_{MMELC} = \mathbb{E}_{\underline{A}|\underline{r}}\left[\underline{Z}\right]. \tag{G.20}$$

Finally, as $N \to \infty$, the MMELC estimator $\underline{\hat{a}}_{MMELC}$ is given by

$$\begin{aligned}
\lim_{N\to\infty} \ \underline{\hat{a}}_{MMELC} &= \ \mathbb{E}_{\underline{A}|\underline{r}}\left[\underline{Z}\right] \\
&= \mathbb{E}_{\underline{A}|\underline{r}}\left[\underline{\underline{H}}\underline{A}|\underline{r}\right] \\
&= \mathbb{E}_{\underline{A}|\underline{r}}\left[\left(\underline{\underline{\mathbf{I}}}_N - \frac{1}{N}\underline{1}\underline{1}^T\right)\underline{A}|\underline{r}\right] \\
&= \mathbb{E}_{\underline{A}|\underline{r}}\left[\underline{A}|\underline{r} - \frac{1}{N}\underline{1}\underline{1}^T\underline{A}|\underline{r}\right] \\
&= \mathbb{E}_{\underline{A}|\underline{r}}\left[\underline{A}|\underline{r}\right] - \frac{1}{N}\underline{1}\underline{1}^T\mathbb{E}_{\underline{A}|\underline{r}}\left[\underline{A}|\underline{r}\right] \\
&= \underline{\hat{a}}_{MMSE} - \underline{\mu}_{\underline{\hat{a}}_{MMSE}}.
\end{aligned} \tag{G.21}$$

In words, the MMELC estimator, $\underline{\hat{a}}_{MMELC}$, is (asymptotically in N) an affine transformation of the STSA-MMSE estimator $\underline{\hat{a}}_{MMSE}$. In practice, this means that using the STSA-MMSE estimator leads to the same approximate-STOI criterion value as the estimator, $\underline{\hat{a}}_{MMELC}$, derived to maximize this criterion. In other words, applying the traditional STSA-MMSE estimator leads to maximum speech intelligibility as reflected by the STOI estimator.

6 Experimental Design

We now investigate empirically the relationship between the MMELC estimator in Eq. (G.14) and the STSA-MMSE estimator in Eq. (G.11) using an experimental study. As defined in Eq. (G.11), the MMELC estimator is the vector that maximizes the expectation of the ELC cost function given by Eq. (G.10). This expectation, Eq. (5), is defined via an integral of $\rho\left(\underline{a}, \underline{\hat{a}}\right)$ for various realizations of $\underline{a}$ and $\underline{\hat{a}}$, and weighted by the joint PDF $f_{\underline{A},\underline{R}}\left(\underline{a}, \underline{r}\right)$. It is however, well known, that the integral may be approximated (arbitrarily well) as a sum of $\rho\left(\underline{a}, \underline{\hat{a}}\right)$ terms, where realizations of $\underline{a}$ and $\underline{\hat{a}}$ are drawn according to $f_{\underline{A},\underline{R}}\left(\underline{a}, \underline{r}\right)$. This is similar to what a DNN approximates during a standard training process, where a gradient based optimization technique is used to minimize the cost on a representative training set [45]. Therefore, training a DNN, e.g. using stochastic gradient ascent, to maximize Eq. (G.4) may be seen as an approximation of Eq. (G.11), where the approximation becomes more accurate with increasing training set size. From the theoretical arguments presented in Sec. 5, we would therefore expect that, for some sufficiently large N, one would obtain equality in an ELC sense, between a DNN trained to maximize an ELC cost function and one that is trained to minimize the classical STSA-MSE cost function.

To validate this expectation we follow the techniques formalized in Secs. 2 and 3 and train DNNs to estimate gain vectors, $\hat{\underline{g}}_{j,m}$, that we apply to noisy one-third octave band magnitude envelope signals $\underline{r}_{j,m}$, to arrive at enhanced signals $\hat{\underline{a}}_{j,m}$. In principle, any supervised learning model would be applicable for these experiments but considering the universal function approximation capability of DNNs [46], this is our model of choice. We use short-time temporal one-third octave band envelope vectors, as defined in Eq. (G.3), and train multiple DNNs, one for each of the $J = 15$ one-third octave bands, for various N, to investigate if for sufficiently large N, DNNs trained with a STSA-MSE cost function approach the ELC values of DNNs trained with a cost function based on ELC.

We construct two types of enhancement systems, one type is trained using the STSA-MSE cost function, denoted as ES_{MSE}, and one that is trained using the ELC cost function denoted as ES_{ELC}. Each of the systems consists of $J = 15$ DNNs, each estimating a gain vector $\hat{\underline{g}}_{j,m}$ for a particular one-third octave band directly from the STFT magnitudes of the noisy signal $r(k, m)$, with the input context given by $k = 1, \ldots, \frac{K}{2} + 1$, $m - N + 1 \ldots, m$. This ensures that all DNNs have access to the same information for a particular value of N, as they all receive the same input data. Furthermore, as $\hat{\underline{g}}_{j,m}$ is estimated for successive values of m, we follow common practice (e.g. [6, 7, 16, 23]) and average overlapping gain values during enhancement.

To compute the STFT coefficients for all signals we use a 10 kHz sam-

ple frequency and a $K = 256$ point STFT with a Hann-window size of 256 samples (25.6 ms) and a 128 sample frame shift (12.8 ms). These coefficients are then used to compute one-third octave band envelopes for the clean and noisy signals using Eq. (G.3).

6.1 Noise-Free Speech Mixtures

We have used the Wall Street Journal (WSJ0) speech corpus [47] as the the clean speech data for both the training set, validation set, and test set. Specifically, the noise-free utterances used for training and validation are generated by randomly selecting utterances from 44 male and 47 female speakers from the WSJ0 training set entitled si_tr_s. In total 20000 utterances are used for the training set and 2000 are used for the validation set, which adds up to approximately 37 hours of training data and 4 hours of validation data. For the test set, we have used a similar approach and sampled 1000 utterances among 16 speakers (10 males and 6 females) from the WSJ0 validation set si_dt_05 and evaluation set si_et_05, which is equivalent to approximately 2 hours of data, see [48] for further details. The speakers used in the training and validation sets are different than the speakers used for test, i.e. we test in a speaker independent setting.

6.2 Noise Types

To simulate a wide variety of sound scenes we have used six different noise types in our experiments: two synthetic noise signals and four natural noise signals, which are real-life recordings of naturally occurring sound scenes. For the two synthetic noise signals, we use a stationary Speech Shaped Noise (SSN) signal and a highly non-stationary 6-speaker babble (BBL) noise. For the naturally occurring noise signals, we use the street (STR), cafeteria (CAF), bus (BUS), and pedestrian (PED) noise signals from the CHiME3 dataset [49]. The SSN noise signal is Gaussian white noise, spectrally shaped according to the long-term spectrum of the entire TIMIT speech corpus [50]. Similarly, the BBL noise signal is constructed by mixing utterances from both genders from TIMIT. To ensure that all noise types are equally represented and with unique realizations in the training, validation and test sets, all six noise signals are split into non-overlapping segments such that 40 min. is used for training, 5 min. is used for validation and another 5 min. is used for test.

6.3 Noisy Speech Mixtures

To construct the noisy speech signals used for training, we follow Eq. (G.1) and combine a noise-free training utterance $x[n]$ with a randomly selected noise sequence $v[n]$, of equal length, from the training noise signal. We scale

the noise signal $v[n]$, to achieve a certain Signal-to-Noise Ratio (SNR), according to the active speech level of $x[n]$ as defined by ITU P.56 [51]. For the training and validation sets, the SNRs are chosen uniformly from $[-5, 10]$ dB to ensure that the intelligibility of the noisy speech mixtures $y[n]$ ranges from degraded to perfectly intelligible.

6.4 Model Architecture and Training

The two types of enhancement systems, ES_{ELC} and ES_{MSE}, each consist of 15 feed-forward DNNs. The DNNs in the ES_{ELC} system are trained with the ELC cost function introduced in Eq. (G.4) and the DNNs in the ES_{MSE} system are trained using the well-known STSA-MSE cost function given by

$$\mathcal{J}(\underline{a}, \hat{\underline{a}}) = \frac{1}{N} \|\underline{a} - \hat{\underline{a}}\|^2, \tag{G.22}$$

where the subscripts j and m are omitted for convenience. We train both the ES_{ELC} and ES_{MSE} systems with 20000 training utterances and 2000 validation utterances and both data sets have been mixed uniformly with the SSN, BBL, CAF, and STR noise signals, which ensures that each noise type have been mixed with 25% of the utterances in the training and validation sets. During test, we evaluate each system with one noise type at a time, i.e. each system is evaluated with 1000 noisy test utterances per noise type, and since BUS and PED are not included in the training and validation sets, these two noise signals serve as unmatched noise types, whereas SSN, BBL, CAF, and STR are matched noise types. This will allow us to study how the ELC optimal DNNs and STSA-MSE optimal DNNs generalize to unmatched noise types.

Each feed-forward DNN consists of three hidden layers with 512 units using ReLU activation functions. The N-dimensional output layer uses sigmoid functions which ensures that the output gain $\hat{\underline{g}}_{j,m}$ is confined between zero and one. The DNNs are trained using stochastic gradient de-/ascent with the backpropagation technique and batch normalization [45]. The DNNs are trained for a maximum of 200 epochs with a minibatch size of 256 randomly selected short-time temporal one-third octave band envelope vectors.

Since the ES_{ELC} and ES_{MSE} systems use different cost functions, they likely have different optimal learning rates. This is easily seen from the gradient norms of the two cost functions. It can be shown (details omitted due to space limitations) that the ℓ^2-norm of the gradient of the ELC cost function in Eq. (G.4), with respect to the desired signal vector $\hat{\underline{a}}$, is given by

$$\|\nabla \mathcal{L}(\underline{a}, \hat{\underline{a}})\| = \frac{\sqrt{1 - \mathcal{L}(\underline{a}, \hat{\underline{a}})^2}}{\|\hat{\underline{a}}\|}, \tag{G.23}$$

where the gradient $\nabla\mathcal{L}(\underline{a},\underline{\hat{a}})$ is given by

$$\nabla\mathcal{L}(\underline{a},\underline{\hat{a}}) = \left[\frac{\partial\mathcal{L}(\underline{a},\underline{\hat{a}})}{\partial\hat{a}_1},\frac{\partial\mathcal{L}(\underline{a},\underline{\hat{a}})}{\partial\hat{a}_2},\ldots,\frac{\partial\mathcal{L}(\underline{a},\underline{\hat{a}})}{\partial\hat{a}_N}\right]^T, \tag{G.24}$$

and

$$\frac{\partial\mathcal{L}(\underline{a},\underline{\hat{a}})}{\partial\hat{a}_m} = \frac{\mathcal{L}(\underline{a},\underline{\hat{a}})\left(\underline{a}_m-\mu_{\underline{a}}\right)}{\left(\underline{\hat{a}}-\mu_{\underline{\hat{a}}}\right)^T\left(\underline{a}-\mu_{\underline{a}}\right)} - \frac{\mathcal{L}(\underline{a},\underline{\hat{a}})\left(\hat{\underline{a}}_m-\mu_{\underline{\hat{a}}}\right)}{\left(\underline{\hat{a}}-\mu_{\underline{\hat{a}}}\right)^T\left(\underline{\hat{a}}-\mu_{\underline{\hat{a}}}\right)}. \tag{G.25}$$

is the partial derivative of $\mathcal{L}(\underline{a},\underline{\hat{a}})$ with respect to entry m of vector $\underline{\hat{a}}$. Similarly, the gradient of the STSA-MSE cost function in Eq. (G.22) is given by

$$\nabla\mathcal{J}(\underline{a},\underline{\hat{a}}) = \frac{2}{N}\left(\underline{a}-\underline{\hat{a}}\right), \tag{G.26}$$

such that

$$\left\|\nabla\mathcal{J}(\underline{a},\underline{\hat{a}})\right\| = \frac{2}{N}\left\|\underline{a}-\underline{\hat{a}}\right\|. \tag{G.27}$$

Note, since $\mathcal{L}(\underline{a},\underline{\hat{a}})$ is invariant to the magnitude of $\|\underline{\hat{a}}\|$ (see Eq. (G.4)), and $\underline{a}$ and N are constants during training, the gradient norm of the ELC cost function, Eq. (G.23), with respect to $\underline{\hat{a}}$, is inversely proportional to the gradient norm of the STSA-MSE cost function, Eq. (G.27). This suggests that the two cost functions have different optimal learning rates. This observation might partly explain why equality with respect to STOI between STOI optimal and STSA-MSE optimal DNNs were achieved in [23] but not in [24–26], as [23] was the only study that explicitly stated that different learning rates for the two cost functions were used. In fact, in [24–26] the optimization method Adam [52] was used, and although Adam is an adaptive gradient method, it still has several critical hyper-parameters that can influence convergence [53].

During a preliminary grid-search using the validation set corrupted with SSN at an SNR of 0 dB and $N = 30$, we found learning rates of 0.01 and $5 \cdot 10^{-5}$ per sample to be optimal for the ES_{ELC} and ES_{MSE} systems, respectively. During training, the cost on the validation set was evaluated for each epoch and the learning rates were scaled by 0.7, if the cost increased compared to the cost for the previous epoch. The training was terminated, if the learning rate was below 10^{-10}. We implemented the DNNs using CNTK [54] and the scripts needed to reproduce the reported results can be found in [48].

7 Experimental Results

To study the relationship between ES_{ELC} and ES_{MSE} systems as function of N, we have trained multiple systems for various N. Specifically, a total of eight ES_{ELC} systems and eight ES_{MSE} systems have been trained with N taking the values $N = \{4, 7, 15, 20, 30, 40, 50, 80\}$, which correspond to temporal envelope vectors with durations from approximately 50 to 1000 milliseconds.

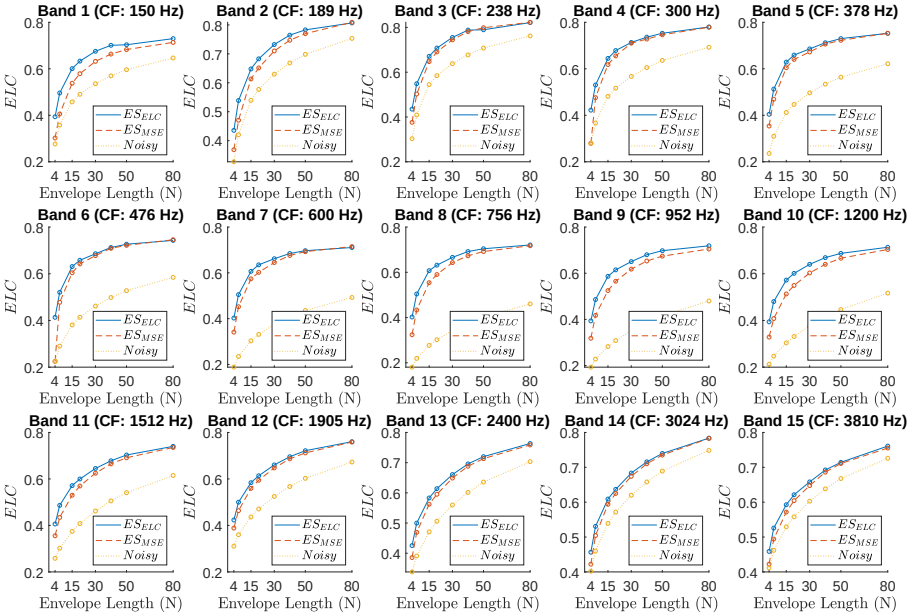


Fig. G.2: ELC values for ES_{ELC} and ES_{MSE} systems trained using various envelope durations, N, and tested with corresponding values of N using speech corrupted with BBL noise at an SNR of 0 dB. Each figure shows one out of $J = 15$ one-third octave band DNNs (center frequency (CF) shown in parenthesis). It is seen that as $N \rightarrow 80$ the difference between the ES_{ELC} and ES_{MSE} DNNs, as measured by ELC, tends to zero. This is in line with the theoretical results of Sec. 5.

7.1 Comparing One-third Octave Bands

In Fig. G.2 we present the ELC scores, as function of envelope duration N, for each of the $J = 15$ one-third octave band DNNs in the ES_{ELC} and ES_{MSE} systems. All DNNs are tested using speech corrupted with BBL noise at an SNR of 0 dB. First, we observe that both systems manage to improve the ELC score considerably, when compared to the ELC score of the noisy speech signals, i.e. both systems enhance the noisy speech, which is in line with known results [8].

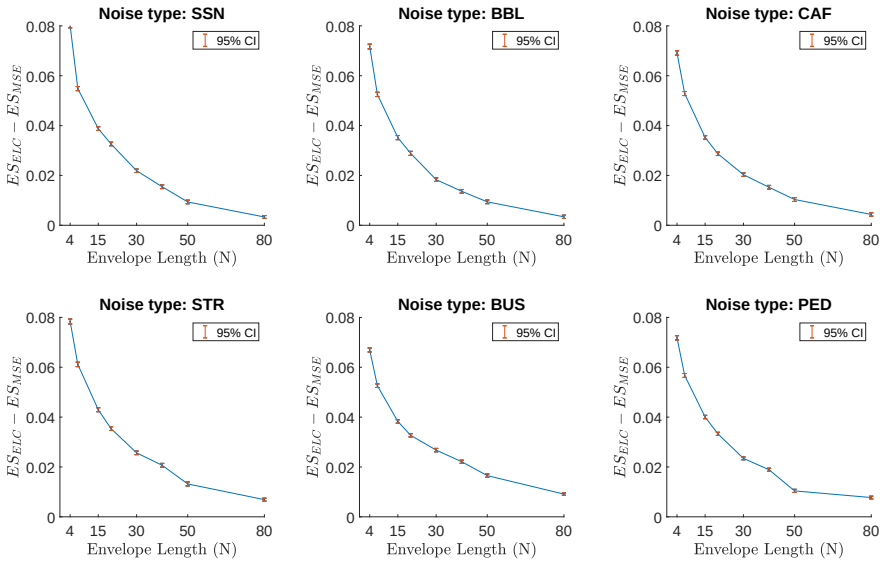


Fig. G.3: Average ELC differences, as function of envelope durations N, between ES_{ELC} and ES_{MSE} systems, for different noise types. We observe a monotonic decreasing relationship between the average ELC difference and the envelope length and for $N = 80$, the average ELC difference between the ES_{ELC} and ES_{MSE} systems is close to zero. This is in line with the theoretical results of Sec. 5.

Furthermore, we can observe that the DNNs trained with the ELC cost function, i.e. the ES_{ELC} systems, in general achieve higher, or similar, ELC scores than the DNNs trained with the STSA-MSE cost function, i.e. the ES_{MSE} systems. This is an important observation, since it verifies that DNNs trained to maximize ELC indeed achieve the highest, or similar, ELC scores compared to DNNs trained to optimize a different cost function, STSA-MSE in this case. Finally, and most importantly, we observe that the difference in ELC score between the ES_{ELC} and ES_{MSE} DNNs generally decrease with increasing N. For $N = 80$ the ELC score of the ES_{ELC} and ES_{MSE} DNNs practically coincide.

7.2 Comparing ELC across Noise Types

In Fig. G.3 we present the ELC score difference, as function of envelope duration N, for ES_{ELC} and ES_{MSE} systems, when tested using speech material corrupted with various noise types at an SNR of 0 dB. Specifically, we compute the difference in ELC score for each pair of one-third octave band DNNs in the ES_{ELC} and ES_{MSE} systems, and then compute the average ELC difference as function of envelope duration N. We do this for all the 1000 test utterances and for each of the six noise types introduced in Sec 6.2: SSN, BBL, CAF, STR,

BUS, and PED. Finally, we compute the 95% confidence interval (CI) on the mean ELC difference.

From Fig. G.3 we observe that the average ELC difference, i.e. $\text{ES}_{ELC} - \text{ES}_{MSE}$, appears to be monotonically decreasing with respect to the duration of the envelope N. Furthermore, we observe that the average ELC difference approaches zero as the duration of the envelope N increases, and similarly to Fig. G.2, for $N = 80$, the difference between the ES_{ELC} and ES_{MSE} systems is close to zero. Finally, we observe that the 95% confidence intervals are relatively narrow for all envelope durations and noise types, which indicate that our test set is sufficiently large to provide accurate estimates of the true mean ELC difference. Similarly to Fig. G.2, the results in Fig. G.3 support the theoretical results of Sec. 5. Additionally, the results in Fig. G.3 show consistency across multiple noise types, which suggests that the theory in practice applies for various noise type distributions.

7.3 Comparing STOI across Noise Types

We now investigate if the global behavior observed for approximate-STOI, i.e. ELC, in Fig. G.3 also applies for real STOI. To do this, we reconstruct the test signals used for Fig. G.3 in the time domain. We follow the technique proposed in [23], where a uniform gain across STFT coefficients within a one-third octave band is used before an inverse DFT is applied using the phase of the noisy signal. In Table G.1 we present the STOI scores for ES_{ELC} and ES_{MSE} systems, as a function of N, when tested using speech material corrupted with different noise types at an SNR of 0 dB. Note that these test signals are similar to the test signals used for Fig. G.3 except that we now evaluate them according to STOI and not ELC.

From Table G.1 we observe that the average STOI difference between the ES_{ELC} and ES_{MSE} systems is maximum for $N = 4$, but quickly tends to zero as N increases and for $N \geq 15$, the STOI difference is practically zero, i.e. ≤ 0.01. Furthermore, we observe that the ES_{MSE} achieve slightly higher STOI scores than the ES_{ELC} systems for $N = 4$, although the maximum improvement in STOI is achieved for $N = \{15, 20, 30\}$, where both systems achieve similar STOI scores. Finally, while the theoretical results of Sec. 5 show that approximate-STOI performance of $\hat{\underline{a}}_{MMELC}$ and $\hat{\underline{a}}_{MMSE}$ is identical, asymptotically, for $N \to \infty$, the empirical results in Table G.1 suggest that $N \geq 15$ is sufficient for practical equality to hold for DNN based speech enhancement systems.

7.4 Comparing Gain-Values

Figures G.2 and G.3, and Table G.1 show that ES_{ELC} systems achieve approximately the same ELC and STOI values as ES_{MSE} systems and that the ELC

Table G.1: STOI scores as function of N for ES_{ELC} and ES_{MSE} systems tested using different noise types at an SNR of 0 dB.

N :		4	7	15	20	30	40	50	80
SSN:	ELC :	0.81	0.85	0.88	0.88	0.87	0.86	0.85	0.84
	MSE :	0.84	0.87	0.87	0.87	0.87	0.86	0.85	0.84
BBL:	ELC :	0.77	0.80	0.82	0.82	0.81	0.80	0.80	0.78
	MSE :	0.79	0.82	0.82	0.82	0.81	0.80	0.80	0.78
CAF:	ELC :	0.82	0.85	0.87	0.87	0.86	0.85	0.84	0.83
	MSE :	0.85	0.87	0.87	0.87	0.86	0.85	0.85	0.84
STR:	ELC :	0.83	0.86	0.88	0.89	0.88	0.87	0.87	0.85
	MSE :	0.86	0.88	0.88	0.88	0.88	0.87	0.87	0.85
PED:	ELC :	0.77	0.81	0.83	0.83	0.83	0.82	0.81	0.80
	MSE :	0.80	0.82	0.83	0.83	0.82	0.82	0.81	0.80
BUS:	ELC :	0.87	0.89	0.90	0.91	0.90	0.89	0.89	0.89
	MSE :	0.89	0.90	0.90	0.90	0.90	0.90	0.89	0.89

and STOI difference between the two types of systems approach zero as N becomes large. These empirical results are in line with the theoretical results in Sec. 5. However, the results in Sec. 5 predict that not only do ES_{ELC}, and ES_{MSE} systems produce identical ELC scores, they also predict that the systems are, in fact, essentially identical, i.e. up to an affine transformation. Hence, in this section, we compare how the systems actually operate. Specifically, we compare the gains estimated by ES_{ELC} systems with gains estimated by ES_{MSE} systems.

In Fig. G.4 we present scatter plots, one for each one-third octave band for pairs of gains estimated by ES_{ELC} and ES_{MSE} systems tested with BBL noise at an SNR of 5 dB. Each scatter plot consists of 10000 pairs of gains acquired by sampling 10 gain-pairs randomly and uniformly distributed from each of the 1000 test utterances. In Fig. G.4, yellow indicates high density of gain-pairs and dark blue indicates low density. From Fig. G.4 it is seen that a correlation no smaller than 0.88 is achieved for all 15 one-third octave bands. The highest correlation of $r = 0.98$ is achieved by bands 5 to 7 and the lowest is $r = 0.88$ achieved by band 2 followed by band 1 with $r = 0.89$. It is also seen that a large number of gain values are either zero, or one, as one would expect due to the sparse nature of speech in the T-F domain. However, although a strong correlation is observed for all bands, the gain-pairs are slightly more scattered at the first few bands than for the remaining bands. This might be explained simply by the fact that low one-third octave bands correspond to single STFT bins, whereas higher one-third octave bands are sums of a large number of STFT bins. This, in turn, may have the consequence that for finite N ($N = 30$), Kolmogorovs strong law of large numbers (see Appendix. B)

Table G.2: Sample correlations between gains from ES_{ELC} and ES_{MSE} systems with $N = 30$. See Fig. G.4 for per band correlations.

SNR [dB]	SSN	BBL	CAF	STR	BUS	PED
-5	0.94	0.87	0.89	0.93	0.87	0.90
0	0.94	0.92	0.92	0.93	0.88	0.92
5	0.95	0.95	0.93	0.93	0.90	0.92
10	0.95	0.95	0.92	0.92	0.91	0.93

is better valid at higher frequencies than at lower frequencies (so that gain vectors produced by one system is closer to an affine transformation of gain vectors produced by the other system). In fact, if we compute r_1 for models trained with $N = 50$, we get $r_1 = 0.93$, i.e. increased correlation between the gain vectors produced by the two systems. Finally, in Table. G.2 we present average correlation coefficients and we observe correlation coefficients ≥ 0.87 for all, both matched and unmatched, noise types, at multiple SNRs.

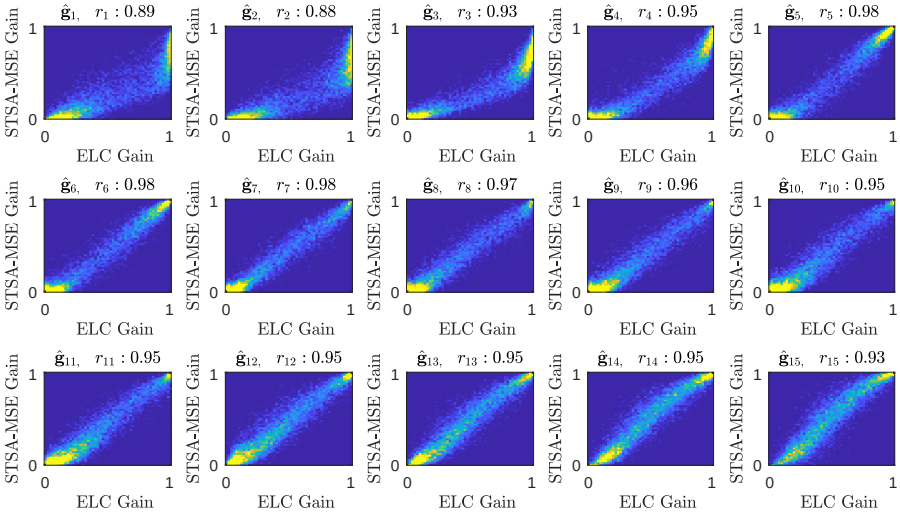


Fig. G.4: Scatter plots based on gain values from ES_{ELC} and ES_{MSE} systems with an envelope length of $N = 30$. Dark blue indicate low density and bright yellow indicate high density. The systems are tested with BBL noise corrupted speech at an SNR of 5 dB. Each figure shows one of 15 ($\hat{\mathbf{g}}_1, \hat{\mathbf{g}}_2, \ldots, \hat{\mathbf{g}}_{15}$) one-third octave bands. A correlation no smaller than 0.88 is achieved for all one-third octave bands, which indicates that the ES_{ELC} and ES_{MSE} systems estimate fairly similar gain vectors.

8 Conclusion

This study is motivated by the fact that most estimators used for speech enhancement, being either data-driven models, e.g. Deep Neural Networks (DNNs), or statistical model-based techniques such as the Short-Time Spectral Amplitude (STSA) Minimum Mean Squared Error (MMSE) estimator, use the STSA Mean Squared Error (MSE) cost function as a performance indicator. Short-Time Objective Intelligibility (STOI), a state-of-the-art speech intelligibility estimator, on the other hand, rely on the Envelope Linear Correlation (ELC) of speech temporal envelopes. Since the primary goal of many speech enhancement systems is to improve speech intelligibility, it raises the question if estimators can benefit from an ELC cost function.

In this paper we derive the Maximum Mean Envelope Linear Correlation (MMELC) estimator and study its relationship to the well-known STSA-MMSE estimator. We show that the MMELC estimator, under a commonly used conditional independence assumption, is asymptotically equivalent to the STSA-MMSE estimator. Furthermore, we show that a similar correspondence holds with respect to STOI for DNN based speech enhancement systems when the DNNs are trained to either maximize ELC or minimize MSE. Our findings suggest, that applying the traditional STSA-MMSE estimator on noisy speech signals in practice leads to maximum speech intelligibility as reflected by the STOI estimator. The important implication of these results is that the classical STSA-MMSE estimator is essentially optimal, if the objective is to achieve optimal performance with respect to a state-of-the-art speech intelligibility estimator.

A Maximizing a Constrained Inner Product

This appendix derives an expression for the zero-mean, unit-norm vector $\underline{e}(\underline{\hat{a}})$, which maximizes the inner product with the vector $\mathbb{E}_{\underline{A}|\underline{r}}[\underline{e}(\underline{A}|\underline{r})]$. For notational convenience, let $\underline{\alpha} = \mathbb{E}_{\underline{A}|\underline{r}}[\underline{e}(\underline{A}|\underline{r})]$, and $\underline{\beta} = \underline{e}(\underline{\hat{a}})$. The constrained optimization problem from Eq. (G.11) is then defined as

$$\begin{aligned} \underset{\underline{\beta}}{\text{maximize}} \quad & \underline{\alpha}^T \underline{\beta} \\ \text{subject to} \quad & \underline{\beta}^T \underline{1} = 0, \\ & \underline{\beta}^T \underline{\beta} = 1. \end{aligned} \tag{G.28}$$

The vector $\underline{\beta}^*$ that solves Eq. (G.28) can be found using the method of Lagrange multipliers [55]. Introducing two scalar Lagrange multipliers, λ_1 and λ_2, for the two equality constraints, the Lagrangian is given by[3]

$$\mathcal{L}(\underline{\beta}, \lambda_1, \lambda_2) = -\underline{\alpha}^T \underline{\beta} + \lambda_1 \underline{\beta}^T \underline{1} + \lambda_2 (\underline{\beta}^T \underline{\beta} - 1). \tag{G.29}$$

Setting the partial derivatives $\frac{\partial \mathcal{L}}{\partial \underline{\beta}}$ equal to zero

$$\frac{\partial \mathcal{L}}{\partial \underline{\beta}} = -\underline{\alpha} + \lambda_1 \underline{1} + 2\lambda_2 \underline{\beta} = \underline{0}, \tag{G.30}$$

and solving for $\underline{\beta}$, we arrive at

$$\underline{\beta} = \frac{\underline{\alpha} - \lambda_1 \underline{1}}{2\lambda_2}. \tag{G.31}$$

Using the same approach for $\frac{\partial \mathcal{L}}{\partial \lambda_1}$ and $\frac{\partial \mathcal{L}}{\partial \lambda_2}$, substituting in Eq. (G.31) and solving for λ_1, and λ_2 such that the two constraints are fulfilled, we find

$$\lambda_1 = \frac{1}{N} \underline{\alpha}^T \underline{1} = \mu_{\underline{\alpha}}, \tag{G.32}$$

and

$$\lambda_2 = \frac{\|\underline{\alpha} - \mu_{\underline{\alpha}} \underline{1}\|}{2}. \tag{G.33}$$

Inserting λ_1 and λ_2 into Eq. (G.31) results in

$$\underline{\beta}^* = \frac{\underline{\alpha} - \mu_{\underline{\alpha}} \underline{1}}{\|\underline{\alpha} - \mu_{\underline{\alpha}} \underline{1}\|}, \tag{G.34}$$

which is simply the vector $\underline{\alpha}$, normalized to zero sample mean and unit norm.

[3] We solve the equivalent problem that minimizes $-\underline{\alpha}^T \underline{\beta}$.

B Factorization of Expectation

This appendix shows that the expectation in Eq. (11) factorizes into the product of expectations in Eq. (G.18), asymptotically as $N \to \infty$. Let

$$\underline{Y} \triangleq \underline{A}|\underline{r}, \tag{G.35}$$

and

$$\underline{\underline{H}} \triangleq \underline{\underline{I}}_N - \frac{1}{N}\underline{1}\underline{1}^T, \tag{G.36}$$

so that

$$\underline{Z} = \underline{\underline{H}}\underline{Y}, \tag{G.37}$$

where $\underline{\underline{I}}_N$ denotes the N-dimensional identity matrix and $\underline{A}|\underline{r}$ is a random vector distributed according to the conditional probability density function $f_{\underline{A}|\underline{R}}(\underline{a}|\underline{r})$. A specific element Z_i, of $\underline{Z}$ is then given by

$$\begin{aligned} Z_i &= \underline{h}_i^T \underline{Y} \\ &= S_i - \frac{1}{N}\underline{1}^T \underline{Y}, \end{aligned} \tag{G.38}$$

where $\underline{h}_i$ is the i^{th} column of matrix $\underline{\underline{H}}$.

We now define the covariance between Z_i and $1/\|\underline{Z}\|$ as

$$\begin{aligned} \operatorname{cov}(Z_i, \frac{1}{\|\underline{Z}\|}) &\triangleq \mathbb{E}\left[\left(Z_i - \mathbb{E}\left[Z_i\right]\right)\left(\frac{1}{\|\underline{Z}\|} - \mathbb{E}\left[\frac{1}{\|\underline{Z}\|}\right]\right)\right] \\ &= \mathbb{E}\left[\frac{Z_i}{\|\underline{Z}\|}\right] - \mathbb{E}\left[Z_i\right]\mathbb{E}\left[\frac{1}{\|\underline{Z}\|}\right]. \end{aligned} \tag{G.39}$$

We can rewrite the factors on the right-hand side of Eq. (G.39) as follows

$$\begin{aligned} \mathbb{E}\left[Z_i\right] &= \mathbb{E}\left[\underline{\mathbf{h}}_i^T \underline{Y}\right] \\ &= \mathbb{E}\left[S_i - \frac{1}{N}\underline{1}^T \underline{Y}\right] \\ &= \mathbb{E}\left[S_i\right] - \frac{1}{N}\underline{1}^T \mathbb{E}\left[\underline{Y}\right] \\ &= \mathbb{E}\left[S_i\right] - \frac{1}{N}\sum_{j=1}^{N}\mathbb{E}\left[S_j\right], \end{aligned} \tag{G.40}$$

$$
\begin{aligned}
\mathbb{E}\left[\frac{1}{\|\underline{Z}\|}\right] &= \mathbb{E}\left[\frac{1}{\sqrt{\underline{Y}^T \underline{\underline{H}}\,\underline{\underline{H}}^T \underline{Y}}}\right] \\
&= \mathbb{E}\left[\frac{1}{\sqrt{\underline{Y}^T \underline{\underline{H}}\,\underline{Y}}}\right] \\
&= \mathbb{E}\left[\frac{1}{\sqrt{\underline{Y}^T \underline{Y} - \frac{1}{N}\underline{Y}^T \underline{1}\underline{1}^T \underline{Y}}}\right] \\
&= \mathbb{E}\left[\frac{1}{\sqrt{\sum_{j=1}^{N} S_j^2 - \frac{1}{N}\left(\sum_{j=1}^{N} S_j\right)^2}}\right] \\
&= \mathbb{E}\left[\frac{\sqrt{\frac{1}{N}}}{\sqrt{\frac{1}{N}\sum_{j=1}^{N} S_j^2 - \left(\frac{1}{N}\sum_{j=1}^{N} S_j\right)^2}}\right],
\end{aligned}
\tag{G.41}
$$

and

$$
\mathbb{E}\left[\frac{Z_i}{\|\underline{Z}\|}\right] = \mathbb{E}\left[\frac{\sqrt{\frac{1}{N}}\left(S_i - \frac{1}{N}\sum_{j=1}^{N} S_j\right)}{\sqrt{\frac{1}{N}\sum_{j=1}^{N} S_j^2 - \left(\frac{1}{N}\sum_{j=1}^{N} S_j\right)^2}}\right]. \tag{G.42}
$$

In Eqs. (G.40), (G.41) and (G.42) two different sums of random variables occur,

$$
\frac{1}{N}\sum_{j=1}^{N} S_j, \tag{G.43}
$$

and

$$
\frac{1}{N}\sum_{j=1}^{N} S_j^2. \tag{G.44}
$$

Since, by assumption, Eq. (G.17), $S_j \ \forall \ j$ are independent random variables with finite variances[4], according to Kolmogorovs strong law of large numbers [44], the sums given by Eqs. (G.43) and (G.44) will converge (almost surely, i.e. with probability (Pr) one) to their average means $\mu_S = \frac{1}{N}\sum_{j=1}^{N} \mathbb{E}[S_j]$, and

[4]Assuming a finite variance of S_j is motivated by the fact that S_j model speech signals, which always take finite values due to both physical and physiological limitations of sound and speech production systems, respectively.

$\mu_{S^2} = \frac{1}{N}\sum_{j=1}^{N}\mathbb{E}[S_j^2]$, respectively, as $N \to \infty$. Formally, we can express this as

$$\Pr\left(\lim_{N\to\infty}\frac{1}{N}\sum_{j=1}^{N}S_j = \mu_S\right) = 1, \tag{G.45}$$

and

$$\Pr\left(\lim_{N\to\infty}\frac{1}{N}\sum_{j=1}^{N}S_j^2 = \mu_{S^2}\right) = 1. \tag{G.46}$$

By substituting Eqs. (G.45), and (G.46) into Eqs. (G.40), (G.41) and (G.42), we arrive at

$$\lim_{N\to\infty}\mathbb{E}\left[Z_i\right] = \mathbb{E}\left[S_i\right] - \mu_S, \tag{G.47}$$

$$\lim_{N\to\infty}\mathbb{E}\left[\frac{1}{\|\underline{Z}\|}\right] = \frac{\lim_{N\to\infty}\sqrt{\frac{1}{N}}}{\sqrt{\mu_{S^2}-\mu_S^2}}, \tag{G.48}$$

and

$$\begin{aligned}\lim_{N\to\infty}\mathbb{E}\left[\frac{Z_i}{\|\underline{Z}\|}\right] &= \left(\mathbb{E}\left[S_i\right] - \mu_S\right)\frac{\lim_{N\to\infty}\sqrt{\frac{1}{N}}}{\sqrt{\mu_{S^2}-\mu_S^2}} \\ &= \lim_{N\to\infty}\mathbb{E}\left[Z_i\right]\mathbb{E}\left[\frac{1}{\|\underline{Z}\|}\right],\end{aligned} \tag{G.49}$$

where the last line follows from Eq. (G.47) and (G.48). In words, as $N \to \infty$, the covariance between Z_i and $1/\|\underline{Z}\|$ tends to zero and, consequently, the expectation in Eq. (11) factorizes into the product of expectations in Eq. (G.18).

References

[1] H. Erdogan, J. R. Hershey, S. Watanabe, and J. L. Roux, "Deep Recurrent Networks for Separation and Recognition of Single-Channel Speech in Nonstationary Background Audio," in *New Era for Robust Speech Recognition*. Springer, 2017, pp. 165–186.

[2] D. Wang, "Deep learning reinvents the hearing aid," *IEEE Spectrum*, vol. 54, no. 3, pp. 32–37, 2017.

[3] D. Wang and J. Chen, "Supervised Speech Separation Based on Deep Learning: An Overview," *arXiv:1708.07524*, 2017.

[4] M. Kim and P. Smaragdis, "Bitwise Neural Networks for Efficient Single-Channel Source Separation," in *NIPS workshop on ML Audio Sig. Process.*, 2017.

[5] R. Fakoor, X. He, I. Tashev, and S. Zarar, "Reinforcement Learning To Adapt Speech Enhancement to Instantaneous Input Signal Quality," *Proc. NIPS Machine Learning for Audio Signal Processing Workshop*, 2017.

[6] J. Chen, Y. Wang, S. E. Yoho, D. Wang, and E. W. Healy, "Large-scale training to increase speech intelligibility for hearing-impaired listeners in novel noises," *The Journal of the Acoustical Society of America*, vol. 139, no. 5, pp. 2604–2612, 2016.

[7] E. W. Healy, M. Delfarah, J. L. Vasko, B. L. Carter, and D. Wang, "An algorithm to increase intelligibility for hearing-impaired listeners in the presence of a competing talker," *The Journal of the Acoustical Society of America*, vol. 141, no. 6, pp. 4230–4239, 2017.

[8] M. Kolbæk, Z. H. Tan, and J. Jensen, "Speech Intelligibility Potential of General and Specialized Deep Neural Network Based Speech Enhancement Systems," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 1, pp. 153–167, 2017.

[9] J. Schnupp, E. Nelken, and A. King, *Auditory Neuroscience - Making Sense of Sound*. MIT Press, 2011.

[10] B. Moore, *An Introduction to the Psychology of Hearing*. Brill, 2013.

[11] R. D. Patterson, K. Robinson, J. Holdsworth, D. Mckeown, C. Zhang, and M. Allerhand, "Complex sounds and auditory images," in *In Proc. International Symposium on Hearing*, 1992, pp. 429–446.

[12] C. H. Taal, R. C. Hendriks, R. Heusdens, and J. Jensen, "An Algorithm for Intelligibility Prediction of Time-Frequency Weighted Noisy Speech," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 7, pp. 2125–2136, 2011.

[13] T. M. Elliott and F. E. Theunissen, "The Modulation Transfer Function for Speech Intelligibility," *PLOS Computational Biology*, vol. 5, no. 3, 2009.

[14] R. Drullman, J. M. Festen, and R. Plomp, "Effect of temporal envelope smearing on speech reception," *The Journal of the Acoustical Society of America*, vol. 95, no. 2, pp. 1053–1064, 1994.

[15] J. S. Lim and A. V. Oppenheim, "Enhancement and bandwidth compression of noisy speech," *Proceedings of the IEEE*, vol. 67, no. 12, pp. 1586–1604, 1979.

[16] E. W. Healy, S. E. Yoho, J. Chen, Y. Wang, and D. Wang, "An algorithm to increase speech intelligibility for hearing-impaired listeners in novel segments of the same noise type," *The Journal of the Acoustical Society of America*, vol. 138, no. 3, pp. 1660–1669, 2015.

[17] P. C. Loizou, "Speech Enhancement Based on Perceptually Motivated Bayesian Estimators of the Magnitude Spectrum," *IEEE Transactions on Speech and Audio Processing*, vol. 13, no. 5, pp. 857–869, 2005.

[18] R. C. Hendriks, T. Gerkmann, and J. Jensen, "DFT-Domain Based Single-Microphone Noise Reduction for Speech Enhancement: A Survey of the State of the Art," *Synthesis Lectures on Speech and Audio Processing*, vol. 9, no. 1, pp. 1–80, 2013.

[19] L. Lightburn and M. Brookes, "SOBM - a binary mask for noisy speech that optimises an objective intelligibility metric," in *Proc. ICASSP*, 2015, pp. 5078–5082.

[20] W. Han, X. Zhang, G. Min, X. Zhou, and W. Zhang, "Perceptual weighting deep neural networks for single-channel speech enhancement," in *Proc. WCICA*, 2016, pp. 446–450.

[21] P. G. Shivakumar and P. Georgiou, "Perception Optimized Deep Denoising AutoEncoders for Speech Enhancement - Semantic Scholar," in *INTERSPEECH*, 2016, pp. 3743–3747.

[22] Y. Koizumi, K. Niwa, Y. Hioka, K. Kobayashi, and Y. Haneda, "DNN-based source enhancement self-optimized by reinforcement learning using sound quality measurements," in *Proc. ICASSP*, 2017, pp. 81–85.

[23] M. Kolbæk, Z.-H. Tan, and J. Jensen, "Monaural Speech Enhancement using Deep Neural Networks by Maximizing a Short-Time Objective Intelligibility Measure," in *Proc. ICASSP*, 2018, pp. 5059 – 5063.

[24] Y. Zhao, B. Xu, R. Giri, and T. Zhang, "Perceptually Guided Speech Enhancement using Deep Neural Networks," in *Proc. ICASSP*, 2018, pp. 5074–5078.

[25] H. Zhang, X. Zhang, and G. Gao, "Training Supervised Speech Separation System to Improve STOI and PESQ Directly," in *Proc. ICASSP*, 2018, pp. 5374–5378.

[26] S. W. Fu, T. W. Wang, Y. Tsao, X. Lu, and H. Kawai, "End-to-End Waveform Utterance Enhancement for Direct Evaluation Metrics Optimization by Fully Convolutional Neural Networks," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 9, pp. 570 – 1584, 2018.

[27] A. W. Rix, J. G. Beerends, M. P. Hollier, and A. P. Hekstra, "Perceptual evaluation of speech quality (PESQ)-a new method for speech quality assessment of telephone networks and codecs," in *Proc. ICASSP*, vol. 2, 2001, pp. 749–752.

[28] S. Jørgensen, J. Cubick, and T. Dau, "Speech Intelligibility Evaluation for Mobile Phones." *Acustica United with Acta Acustica*, vol. 101, pp. 1016–1025, 2015.

[29] J. Jensen and C. H. Taal, "An Algorithm for Predicting the Intelligibility of Speech Masked by Modulated Noise Maskers," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 11, pp. 2009–2022, 2016.

[30] ——, "Speech Intelligibility Prediction Based on Mutual Information," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 22, no. 2, pp. 430–440, 2014.

[31] T. H. Falk *et al.*, "Objective Quality and Intelligibility Prediction for Users of Assistive Listening Devices: Advantages and limitations of existing tools," *IEEE Signal Processing Magazine*, vol. 32, no. 2, pp. 114–124, 2015.

[32] R. Xia, J. Li, M. Akagi, and Y. Yan, "Evaluation of objective intelligibility prediction measures for noise-reduced signals in mandarin," in *Proc. ICASSP*, 2012, pp. 4465–4468.

[33] P. C. Loizou, *Speech Enhancement: Theory and Practice*. CRC Press, 2013.

[34] Y. Ephraim and D. Malah, "Speech enhancement using a minimum-mean square error short-time spectral amplitude estimator," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 32, no. 6, pp. 1109–1121, 1984.

[35] G. Kim, Y. Lu, Y. Hu, and P. C. Loizou, "An algorithm that improves speech intelligibility in noise for normal-hearing listeners," *The Journal of the Acoustical Society of America*, vol. 126, no. 3, pp. 1486–1494, 2009.

[36] K. Han and D. Wang, "A classification based approach to speech segregation," *The Journal of the Acoustical Society of America*, vol. 132, no. 5, pp. 3475–3483, 2012.

[37] J. Allen, "Short term spectral analysis, synthesis, and modification by discrete Fourier transform," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 25, no. 3, pp. 235–238, 1977.

[38] C. H. Taal, R. C. Hendriks, and R. Heusdens, "Matching pursuit for channel selection in cochlear implants based on an intelligibility metric," in *Proc. EUSIPCO*, 2012, pp. 504–508.

[39] A. H. Andersen, J. M. d. Haan, Z. H. Tan, and J. Jensen, "Predicting the Intelligibility of Noisy and Nonlinearly Processed Binaural Speech," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 11, pp. 1908–1920, 2016.

[40] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*. Prentice Hall, 2010.

[41] Y. Ephraim and D. Malah, "Speech enhancement using a minimum mean-square error log-spectral amplitude estimator," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 33, no. 2, pp. 443–445, 1985.

[42] J. S. Erkelens, R. C. Hendriks, R. Heusdens, and J. Jensen, "Minimum Mean-Square Error Estimation of Discrete Fourier Coefficients With Generalized Gamma Priors," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 6, pp. 1741–1752, 2007.

[43] R. McAulay and M. Malpass, "Speech enhancement using a soft-decision noise suppression filter," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 28, no. 2, pp. 137–145, 1980.

[44] P. K. Sen and J. M. Singer, *Large Sample Methods in Statistics: An Introduction with Applications*. Chapman & Hall, 1994.

[45] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016.

[46] K. Hornik, M. Stinchcombe, and H. White, "Multilayer feedforward networks are universal approximators," *Neural Networks*, vol. 2, no. 5, pp. 359–366, 1989.

[47] J. Garofolo, D. Graff, P. Doug, and D. Pallett, "CSR-I (WSJ0) Complete LDC93s6a," 1993, philadelphia: Linguistic Data Consortium.

[48] M. Kolbæk, Z.-H. Tan, and J. Jensen, "Supplemental Material." [Online]. Available: http://kom.aau.dk/~mok/taslp2018

[49] J. Barker, R. Marxer, E. Vincent, and S. Watanabe, "The third 'CHiME' speech separation and recognition challenge: Dataset, task and baselines," in *Proc. ASRU*, 2015, pp. 504–511.

[50] J. S. Garofolo, L. F. Lamel, W. M. Fisher, J. G. Fiscus, D. S. Pallett, and N. L. Dahlgren, "DARPA TIMIT Acoustic Phonetic Continuous Speech Corpus CDROM," 1993.

[51] ITU, "Rec. P.56 : Objective measurement of active speech level," 1993, https://www.itu.int/rec/T-REC-P.56/.

[52] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," in *Proc. ICLR (arXiv:1412.6980)*, 2014.

[53] A. C. Wilson, R. Roelofs, M. Stern, N. Srebro, and B. Recht, "The Marginal Value of Adaptive Gradient Methods in Machine Learning," in *Proc. NIPS*, 2017.

[54] A. Agarwal *et al.*, "An introduction to computational networks and the computational network toolkit," Microsoft Technical Report {MSR-TR}-2014-112, Tech. Rep., 2014.

[55] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.

ISSN (online): 2446-1628
ISBN (online): 978-87-7210-256-6

AALBORG UNIVERSITY PRESS